การวิเคราะห์ตัวพิมพ์อักษรภาษาไทยโดยใช้ CPN

RESOLVING PRINTED THAI CHARACTERS WITH CPN

T 027112

นายอภิรักษ์ จิรายุสกุล

MR. APIRAK JIRAYUSAKUN

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน 27112
วัน, เดือน, ปี 11 ก.พ. 2540

พ.ศ. 2539

ISBN 974-621-590-6

ลิขสิทธิ์ของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RESOLVING PRINTED THAI CHARACTERS WITH CPN

APIRAK JIRAYUSAKUN

A THESIS SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENTS FOR THE DEGREE

MASTER OF SCIENCE PROGRAM IN COMPUTER SCIENCE

AND INFORMATION TECHNOLOGY

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

1996

ISBN 974-621-590-6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การวิเคราะห์ตัวพิมพ์อักษรภาษาไทยโดยใช้ CPN
นักศึกษา	นายอภิรักษ์ จิรายุสกุล
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ดร.บุญธีร์ เครือตราชู
ระดับการศึกษา	วิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ.	2539

บทคัดย่อ

งานวิจัยนี้ได้นำ Counterpropagation network หรือ CPN มาประยุกต์ใช้ในงานวิเคราะห์ตัวพิมพ์อักษรภาษาไทย โดยมีการเปลี่ยนแปลงโครงสร้างหลักของ CPN 2 ส่วน คือ ส่วนแรกวิธีการเรียนรู้โดยใช้วิธีการแยกสอนตัวอย่างข้อมูลทีละกลุ่ม ส่วนที่สองคือ ให้โครงสร้างในชั้นของคอมเพททิทิพ (Competitive Layer) มีสามารถในการเพิ่มจำนวนของโหนด (node) ในขณะขั้นตอนของการฝึกสอนโครงข่าย ซึ่งวิธีนี้ช่วยลดเวลาในการเรียนรู้และความผิดพลาดในการวิเคราะห์ตัวอักษร

วิทยานิพนธ์ฉบับนี้จะแสดงให้เห็นถึงปัญหาและวิธีการแก้ไขโครงสร้างของ CPN เพื่อนำมาประยุกต์ใช้ในการวิเคราะห์ตัวอักษรภาษาไทย ผลของการทดสอบความสามารถในการวิเคราะห์ตัวพิมพ์อักษรภาษาไทย 75 ตัว ซึ่งใช้ข้อมูลที่พิมพ์ด้วยเครื่องเลเซอร์จำนวน 15,000 ตัว และข้อมูลที่ผ่านการถ่ายเอกสารจำนวน 15,000 ตัว มีความสามารถในการวิเคราะห์ได้ 98.1 % โดยจำนวนตัวอย่างที่นำมาสอนโครงข่ายจำนวน 1,500 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Resolving Printed Thai Characters with CPN
Student	Mr.Apirak Jirayusakun
Thesis Advisor	Dr.Boontee Kruatrachue
Level of Study	Master of Science program in Computer Science and Information Technology
Department	Mathematics and Computer Science Department Faculty of Science King Mongkut's Institute of Technology Ladkrabang
Year	1996

ABSTRACT

This thesis explores the use of counterpropagation neural network (CPN) in a printed Thai character classification. The network is modified in two main parts. The first part is in the learning method where group separation learning is used. The second part,which is the main modified part,is in the network architecture where the number of nodes in the CPN competitive layer can be increased during the learning phase. This will help reduce the error in character recognition.

The problem of CPN in application to Thai character classification and the modification are presented. Then,the new modified network is used to test 75 Thai characters classification from 15,000 laser printed samples and 15.000 photocopy samples with the classification rate up to 98.1 %,using 1.500 training characters.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยดีเพราะ ได้รับความเมตตาจาก ดร.บุญธีร์ เครือตราชู ที่ได้ให้ความกรุณาสละเวลา เพื่อให้คำแนะนำและปรึกษาแนวทางการแก้ปัญหาต่างๆตลอดการทำวิจัยนี้ ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูง

ขอขอบคุณ บิดา,มารดา และพี่สาวที่ให้กำลังใจในการทำวิจัย

ขอขอบคุณ คุณอัญชลี วานิชทวีวัฒน์ ,คุณทวี เปรมรัตนชัย ,คุณดุลยรัชย์ สุขบุญสถิตย์ ที่ช่วยจัดเตรียมข้อมูลต่างๆเพื่อใช้ในการทดลอง

ขอขอบคุณ คุณรุ่งโรจน์ โพนคำ,ปราโมทย์ ปทีปกมล ที่กรุณาให้คำแนะนำเรื่องการเขียนโปรแกรมบนวินโดว์ เพื่อเป็นโปรแกรมที่ใช้ในการทดสอบของงานวิจัยนี้

ขอขอบคุณ คุณสมเกียรติ เหรียญตระกูลชัยและภรรยา,คุณสุวิน ชำนาญช่าง,คุณสรวิศ บุญมี ที่กรุณาช่วยจัดหาอุปกรณ์ในการทำวิจัย

ขอขอบคุณ คุณสุกิจ คูชัยสิทธิ์,สมเกียรติ บริบาลประสิทธิ์,สิทธิชัย บุษหมั่น,ประภัสสร โชคสวนทรัพย์,สิทธิศักดิ์ รอดทุกข์,ณรรฐพงษ์ วงษ์พานิช,พ.ต.ต.บุญชัย ฤาชัยสา,ธงชัย เส็งศรี,ชูศักดิ์ ศิริรุ่งพันธ์,ชัยรัตน์ ฤทธิรงค์,วรรณา จิฐ (เจ๊เจื๋อ) ที่กรุณาให้คำแนะนำในการทำวิทยานิพนธ์ด้วยดีตลอดมา และเพื่อนๆ รุ่น 4 สามัญสาขาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศทุกคนที่เป็นกำลังใจในการทำวิจัยนี้

ขอขอบคุณพี่บุญช่วย และพี่ๆเจ้าหน้าที่คณะเทคโนโลยีสารสนเทศที่ให้การช่วยเหลือด้านต่างๆ

อภิรักษ์ จิรายุสกุล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ.....IV

สารบัญตาราง.....VII

สารบัญภาพ.....X

อภิธานศัพท์.....XIII

บทที่

1. บทนำ.....1

ความเป็นมาของงานวิจัยสาขานี้.....1

ผลงานวิจัยที่ผ่านมา.....2

แนวทางของวิทยานิพนธ์.....4

วัตถุประสงค์ของงานวิจัย.....4

โครงสร้างของวิทยานิพนธ์.....5

2. ทฤษฎีเบื้องต้นของโครงข่ายประสาทเทียม.....6

ตัวแบบของเซลประสาทเทียม.....7

สถาปัตยกรรมของโครงข่ายประสาทเทียม.....11

การฝึกสอนโครงข่าย.....13

การฝึกสอนแบบมีผู้สอน.....12

การฝึกแบบไม่มีผู้สอน.....17

การแบ่งแบบเชิงเส้น.....18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่

3. ทฤษฎีและปัญหาของ CPN ที่ใช้ในการวิเคราะห์ตัวอักษรภาษาไทย.....21
โครงสร้างของโครงข่ายเคาน์เตอร์รอพาเกชัน.....21
หลักการของ CPN ที่ใช้ในการวิจัย.....25
วิธีการสอน CPN 27
การสอน CPN ในชั้นคอมเพททิทิพ.....27
อัลกอลิทึมที่ใช้ในการสอนชั้นคอมเพททิทิพของ CPN.....30
ปัญหาการใช้ CPN ในการแยกแยะตัวพิมพ์อักษรภาษาไทย.....30
ปัญหาการเรียนผิดโหนด.....30
ปัญหาการหาค่าน้ำหนักเฉลี่ยผิด.....33
4. การประยุกต์ CPN เพื่อการแยกแยะตัวพิมพ์อักษรภาษาไทย.....35
การฝึกสอน CPN เพื่อการแยกแยะตัวพิมพ์อักษรภาษาไทย.....36
การกำหนดค่าถ่วงน้ำหนักเริ่มต้น.....36
การกำหนดให้ข้อมูลที่เข้าสู่โครงข่ายอยู่ในรูป +1,-1 43
วิธีการคำนวณค่า net input.....44
วิธีการปรับค่าถ่วงน้ำหนัก.....45
การเพิ่มจำนวนโหนดในชั้นคอมเพททิทิพของ CPN.....46
การแยกสอนของ CPN.....49
สรุปขั้นตอนการสอนโครงข่าย CPN.....54
5. ผลการทดลองการแยกแยะตัวพิมพ์อักษรภาษาไทย.....55
การทดลองการแยกแยะตัวอักษรที่ละชุดตัวอักษร.....55
การทดลองการแยกแยะตัวอักษรที่อยู่ในรูปประโยค.....79

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่

6. บทสรุปและแนวทางการพัฒนาในอนาคต....82
บทสรุป....82
แนวทางการพัฒนาในอนาคต....86
บรรณานุกรม....87
ภาคผนวก....89
ภาคผนวก ก การแยกภาพตัวอักษรออกจากภาพประโยค....90
ภาคผนวก ข เทคนิคการติดตามรอยขอบของภาพ....93
ภาคผนวก ค การจัดเรียงรูปประโยคหลังการวิเคราะห์....94
ภาคผนวก ง บทความวิชาการ....95
ประวัติผู้เขียน....102

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่

1. แสดงข้อมูลที่นำมาสอนเพอร์เซปตรอนและผลลัพธ์ที่ต้องการ.....14
2. แสดงขั้นตอนของเพอร์เซปตรอนจนสามารถแยกข้อมูลที่นำมาสอน.....16
3. แสดงค่าเทรสโฮลด์ที่ใช้ในการสอนโครงข่าย A.....58
4. เปอร์เซ็นต์การเรียนทุก ๆ 4 รอบใช้จำนวนข้อมูลที่นำมาสอน 1500 ตัว ของโครงข่าย A.....59
5. แสดงจำนวนโหนดในชั้นคอมเพททิทิพของแต่ละกลุ่มตัวอักษรของโครงข่าย A.....60
6. แสดงผลการทดสอบการแยกตัวอักษรภาษาไทยชุดละ 15,000 ตัว โดยใช้โครงข่าย A.....60
7. แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษพิมพ์เลเซอร์ที่ความสว่าง 100 เปอร์เซ็นต์ของโครงข่าย A.....61
8. แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษพิมพ์เลเซอร์วามสว่าง 50 เปอร์เซ็นต์ของโครงข่าย A.....62
9. แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษถ่ายเอกสารที่ความสว่าง 100 เปอร์เซ็นต์ของโครงข่าย A.....63
10. แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มตัวอักษรด้วยกระดาษถ่ายเอกสารที่ความสว่าง 50 เปอร์เซ็นต์ของโครงข่าย A.....64
11. แสดงค่าเทรสโฮลด์ที่ใช้ในการสอนโครงข่าย B.....65
12. เปอร์เซ็นต์การเรียนทุก ๆ 4 รอบใช้จำนวนข้อมูลเข้าที่มาสอน 1500 ตัว ของโครงข่าย B.....66

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง (ต่อ)

หน้า

ตารางที่

13. แสดงจำนวนโหนดในชั้นคอมเพททิทิพของแต่ละกลุ่มตัวอักษร ของโครงข่าย B.....67
14. แสดงผลการทดสอบการแยกตัวอักษรภาษาไทยชุดละ 15,000 ตัว โดยใช้โครงข่าย B 68
15. แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษพิมพ์เลเซอร์ ที่ความสว่าง 100 เปอร์เซ็นต์ของโครงข่าย B 68
16. แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มตัวอักษรด้วยกระดาษ พิมพ์เลเซอร์ที่ความสว่าง 50 เปอร์เซ็นต์ของโครงข่าย B 69
17. แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษถ่ายเอกสาร ที่ความสว่าง 100 เปอร์เซ็นต์ของโครงข่าย B 70
18. แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษถ่ายเอกสาร ที่ความสว่าง 50 เปอร์เซ็นต์ของโครงข่าย B.....71
19. แสดงค่าเทรสโฮลด์ที่ใช้ในการสอนโครงข่าย C.....72
20. เปอร์เซ็นต์การเรียนทุก ๆ 4 รอบ ใช้จำนวนข้อมูลเข้าที่มาสอน 1500 ตัว ของโครงข่าย C.....73
21. แสดงจำนวนโหนดในชั้นคอมเพททิทิพของแต่ละกลุ่มตัวอักษร ของโครงข่าย C.....74
22. แสดงผลการทดสอบการแยกตัวอักษรภาษาไทยชุดละ 15000 ตัว ของโครงข่าย C.....74
23. แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษพิมพ์เลเซอร์ ที่ความสว่าง 100 เปอร์เซ็นต์ของโครงข่าย C.....75
24. แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มตัวอักษรด้วยกระดาษ พิมพ์เลเซอร์ที่ความสว่าง 50 เปอร์เซ็นต์ของโครงข่าย C.....76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง (ต่อ)

หน้า

ตารางที่

25. แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษถ่ายเอกสาร ที่ความสว่าง 100 เปอร์เซ็นต์ของโครงข่าย C 77
26. แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษถ่ายเอกสาร ที่ความสว่าง 50 เปอร์เซ็นต์ของโครงข่าย C 78
27. แสดงผลการทดสอบแยกแยะตัวอักษรในประโยค 80

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

หน้า

22. แสดงปัญหาการเรียนผิดโหนดของ CPN กรณีระยะห่างระหว่างขอบเขตของ.....31
23. แสดงปัญหาการเรียนผิดโหนดของ CPN กรณีกำหนดทิศทางเริ่มต้น
ของค่าถ่วงน้ำหนักไม่ดี.....32
24.แสดงปัญหาการหาค่าทิศทางของค่าถ่วงน้ำหนักผิด.....34
25. แสดงการนำข้อมูลภาพตัวอักษรเข้าสู่โครงข่าย CPN ที่ใช้ในงานวิจัย.....35
26. แสดงทิศทางการปรับค่าถ่วงน้ำหนักของการสอนโครงข่ายให้แยกแยะ ก และ ค
โดยใช้การกำหนดเริ่มต้นค่าถ่วงน้ำหนักแบบสุ่ม.....37
27. แสดงทิศทางการปรับค่าถ่วงน้ำหนักของการสอนโครงข่ายให้แยกแยะ ก และ ค
โดยใช้ตัวอย่างข้อมูลเป็นตัวแบบ.....38
28. แสดงปัญหาการกำหนดค่าถ่วงน้ำหนักเริ่มต้นด้วยวิธีการนำตัวอย่างข้อมูลที่ผ่าน
การนอร์มอล์ไลซ์มาต้นแบบ.....41
29. แสดงตัวอย่างเมตริกของค่าถ่วงน้ำหนักเริ่มต้น.....42
30. แสดงการซ้อนทับของโหนดของค่าถ่วงน้ำหนัก ด กับข้อมูล ต ที่เข้าสู่โครงข่าย.....43
31. แสดงเมตริกของอินพุตเวคเตอร์ของ ก ที่เข้าสู่โครงข่าย.....44
32. แสดงขั้นตอนการคิดค่า net input.....45
33. แสดงการใช้วิธีการเพิ่มโหนดในชั้นคอมเพททิทิพเพื่อแก้ปัญหา
การหาค่าถ่วงน้ำหนักเฉลี่ยผิดในภาพที่ 24.....47
34. แสดงการแก้ปัญหาภาพที่ 22. ด้วยวิธีการเพิ่มโหนดในชั้นคอมเพททิทิพ.....48
35. แสดงปัญหาเรียนผิดโหนดกรณีระยะห่างของอินพุตเวคเตอร์ X22 ก่อนการสอน
มีระยะห่างน้องกว่าค่าถ่วงน้ำหนักของกลุ่มข้อมูลที่ 2 ซึ่งเป็นผลให้ W11 ถูกเรียน.....50
36. แสดงตัวอย่างอย่างปัญหาการสอน ก,ฉ กรณีทิศของข้อมูลที่นำมาสอนโครงข่าย
มีระยะไกลจากค่าถ่วงน้ำหนักของกลุ่มข้อมูลที่ต้องการสอน.....51
37. แสดงอธิบายปัญหาภาพที่ 36 ด้วยรูปทิศทางการปรับค่าถ่วงน้ำหนัก.....52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

หน้า

38. แสดงการทิศทางการปรับค่าถ่วงน้ำหนักของภาพที่ 36 และ 37 กรณีแยกสอน.....53
39. แสดงตัวอักษรที่นำมาทดลองการแยกแยะทีละชุดตัวอักษร.....55
40. แสดงการวางตัวอักษรบนเมตริกเพื่อเป็นข้อมูลที่เข้าสู่โครงข่ายที่ใช้ในการทดลอง.....57
41. แสดงตัวอักษรของที่ใช้สอนโครงข่ายบนและล่าง.....80
42. แสดงตัวอย่างภาพของรูปประโยคนำมาใช้ในการทดสอบ.....81
43. แสดงตัวอย่างภาพตัวอักษรที่โครงข่าย CPN สามารถแยกแยะได้.....83
44. แสดงตัวอย่างภาพตัวอักษรที่โครงข่าย CPN ไม่สามารถแยกแยะได้.....84
45. แสดงการตรวจกวาดจุดดำของภาพตามแนวนอนในการแยกภาพตัวอักษร.....90
46. แสดงผลของการตัดบรรทัดด้วยวิธีการทำเบลอ.....91
47. แสดงการแยกภาพคำว่า "รี่" จากภาพประโยค.....92
48. แสดงผลลำดับการดึงภาพตัวอักษรออกจากประโยค.....92
49. แสดงลักษณะการติดตามรอยขอบภาพ.....93

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อภิธานศัพท์

Activation function ฟังก์ชันที่ใช้กำหนดระดับสัญญาณการกระตุ้นที่ออกจากเซลประสาทเทียม

Adaline เซลประสาทเทียมชนิดหนึ่งซึ่งถูกพัฒนาโดย Bernard Widrow ในปี 1960

ART network โครงข่ายประสาทเทียมแบบหนึ่งที่ถูกพัฒนาโดยใช้ทฤษฎี adaptive resonance theory ซึ่งมีทั้งหมด 3 แบบ คือ ART1 ถูกพัฒนาขึ้นเพื่อใช้กับข้อมูลเข้าที่มีลักษณะเป็นแบบค่าไบนารี (binary), ART2 ใช้กับข้อมูลเข้าที่มีลักษณะเป็นเลขจำนวนจริง, ART3 ถูกพัฒนาจาก ART2 เพื่อให้มีความสามารถดีกว่า

BAM โครงข่ายประสาทเทียมแบบหนึ่งที่ประกอบด้วย 2 ชั้น

Backpropagation network โครงข่ายประสาทเทียมชนิดนี้มีโครงสร้างเป็นแบบหลายชั้น (multilayer) ที่ใช้กฎ delta rule ในการปรับค่าถ่วงน้ำหนักหรือการเรียนรู้

Hopfield network เป็นโครงข่ายประสาทเทียมแบบหนึ่งที่โครงสร้างแบบชั้นเดียว (single layer) ซึ่งถูกพัฒนาโดย John Hopfield

Input vector คือข้อมูลที่เข้าสู่ตัวเซลของโครงข่ายประสาทเทียม

Self-Organizing Maps (SOM) network ถูกพัฒนาโดย Teuvo Kohonen ซึ่งในการสอนโครงข่ายจะไม่มีการกำหนดเป้าหมายให้กับโครงข่าย

Kohonen layer คือ ชั้นของโครงข่าย SOM ที่มีการทำงานในลักษณะแบบแข่งขัน

Perceptron โครงข่ายชนิดมีเพียงชั้นเดียว และสามารถแก้ปัญหาได้เพียงปัญหาที่เป็นแบบเชิงเส้นเท่านั้น (linear separability)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

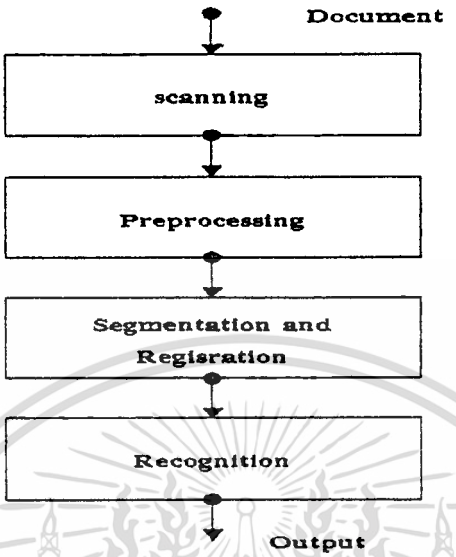
<u>ความเป็นมาของงานวิจัยในสาขานี้</u>

ปัจจุบันเครื่องคอมพิวเตอร์ได้ถูกนำมาประยุกต์ใช้ในวงการต่างๆเกือบทุกสาขาอาชีพ ซึ่งรูปแบบการป้อนข้อมูลให้กับคอมพิวเตอร์ทำด้วยวิธีการพิมพ์ผ่านแป้นพิมพ์หรือคีย์บอร์ด (keyboard) โดยทั่วๆไปแล้ววิธีการนี้เป็นวิธีที่ง่ายและสะดวกสำหรับผู้ที่มีความชำนาญในการพิมพ์ดีด แต่สำหรับผู้ที่ยังขาดความชำนาญแล้วโอกาสผิดพลาดของข้อมูลมีสูงมาก และทั้งยังเสียเวลาในการพิมพ์ด้วยเนื่องจากขาดความชำนาญ แต่ปัญหาที่สำคัญคือ กรณีที่ต้องการจัดเก็บข้อมูลที่มีปริมาณมากๆนั้นการนำข้อมูลเข้าสู่คอมพิวเตอร์โดยผ่านทางแป้นพิมพ์ต้องใช้บุคลากรและอุปกรณ์จำนวนมากซึ่งส่งผลให้ต้องสิ้นเปลืองงบประมาณ ดังนั้นจึงได้มีการพัฒนาเครื่องอ่านตัวอักษรด้วยแสง (Optical Character Reader) มาใช้งาน

กระบวนการของเครื่องอ่านตัวอักษรด้วยแสงแสดงดังภาพที่ 1[1] ประกอบด้วย 4 ขั้นตอน คือ การสแกนซึ่งเป็นกระบวนการแปลงสัญญาณไฟฟ้าให้อยู่ในรูปของดิจิตอล (digital) ซึ่งทำให้เครื่องคอมพิวเตอร์สามารถนำมาประมวลผลได้,กระบวนการประมวลผลส่วนหน้า (preprocessing) เป็นกระบวนการที่เกี่ยวกับการขจัดสิ่งรบกวน (noise) ที่ได้จากการสแกนของภาพและทำการแปลงข้อมูลภาพให้อยู่ในรูปขาวดำ กระบวนการที่สามคือการทำเซกเมนต์ (segmentation) และการลงทะเบียนตัวอักษร (registration) ซึ่งทำหน้าที่ในการแยกภาพตัวอักษรและบันทึกตำแหน่งของภาพเพื่อนำไปใช้ในกระบวนการถัดไป กระบวนการสุดท้ายคือ การจดจำตัวอักษร (recognition) ซึ่งประกอบด้วยสองส่วนย่อยคือ การวิเคราะห์หรือแยกแยะตัวอักษร (classification) ซึ่งส่วนนี้ทำหน้าที่วิเคราะห์ภาพตัวอักษรที่ได้จากกระบวนการที่สามว่าคือตัวอักษรอะไรแล้วแปลงให้เป็นรหัสแอสกี ส่วนย่อยที่สองคือการตรวจสอบว่าตัวอักษรที่ได้จากการวิเคราะห์มีความถูกต้องหรือไม่และจัดเรียงลำดับตัวอักษร ซึ่งการจัดเรียงนี้จะนำตำแหน่งที่บันทึกในขั้นตอนของกระบวนการเซกเมนต์มาจัดลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 1



แสดงกระบวนการของเครื่องอ่านตัวอักษรด้วยแสง

<u>ผลงานวิจัยที่ผ่านมา</u>

ผลงานวิจัยด้านการวิเคราะห์ตัวอักษรภาษาไทยได้มีนักวิจัยผู้มีคุณวุฒิหลายท่านทำการศึกษาค้นคว้าไว้มากมายหลายท่าน ซึ่งพอจะยกตัวอย่างและสรุปวิธีการทำงานพอสังเขปได้ดังนี้

1. ผลงานวิจัยของ รศ.ดร.ชม กิ้มปาน "เรื่องการรู้จำรูปแบบอักษรพิมพ์ภาษาไทยของคอมพิวเตอร์"[10] ซึ่งได้นำเสนอไว้ 2 วิธีคือ

การรู้จำแบบซ้อนทับ (Matching method) วิธีการนี้ใช้หลักการที่ว่ารูปร่างลักษณะของตัวอักษรภาษาไทยมีรูปร่างที่คล้ายกัน ซึ่งสามารถจัดเป็นกลุ่มอย่างคร่าวๆได้ ขั้นตอนการวิเคราะห์แบ่งออกเป็น 2 ขั้นตอนใหญ่ๆคือ ขั้นตอนการวิเคราะห์แบบหยาบ ขั้นตอนนี้ใช้เทคนิคการทำเบลอเพื่อกำจัดรายระเอียดปลีกย่อย และนำทฤษฎีการกระจายแบบคาร์ฮูแนนโลบ (Karhunen Loeve expansion) หารูปแบบมาตราฐานสำหรับเก็บไว้เป็นตัวแทนของกลุ่ม ส่วนขั้นตอนการวิเคราะห์อย่างละเอียดมีสองวิธี วิธีแรกใช้การซ้อนทับเป็นส่วนๆ (subpattern matching) ในการแยกตัวอักษรออกจากัน วิธีที่สองใช้ฟังก์ชันตัดสินใจแบบเชิงเส้น (linear decision functions) บนระนาบของไอเกนเวคเตอร์ที่ได้จากการกระจายแบบคาร์ฮูแนนโลบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวิเคราะห์โครงสร้าง (Structural analysis method) วิธีการนี้นำเอาคุณสมบัติทางโทโพโลยี (Topology) ช่วยในการพิจารณา การแบ่งกลุ่มแบบหยาบของภาพตัวอักษรที่ต้องการวิเคราะห์ ทำโดยการพิจารณาความสูงและความกว้างของตัวอักษร ซึ่งก่อนการวิเคราะห์จะทำภาพตัวอักษรให้บางก่อน จากนั้นจึงนำคุณสมบัติทางโทโพโลยีมาวิเคราะห์ความแตกต่างของตัวอักษร

2. ผลงานวิจัยของ นายประสาร ตังติสานนท์ เรื่อง "การจดจำรูปแบบตัวอักษรคัดลายมือภาษาไทย"[11] ใช้หลักการพิจารณาโครงสร้างของลายเส้นและพื้นเบื้องหลังของลายเส้นของตัวอักษรควบคู่กันไป การวิเคราะห์ตัวอักษร ประกอบด้วย 3 ขั้นตอนใหญ่คือ ขั้นตอนแรกเปลี่ยนลายเส้นและพื้นเบื้องหลังลายเส้นไปเป็นรหัสเบื้องต้น (Initial feature extraction) ขั้นที่สองทำการลดทอนรหัสเบื้องต้นที่ซ้ำซ้อน (Unification) ขั้นตอนสุดท้ายนำรหัสของลายเส้นและพื้นเบื้องหลังตัวอักษรมาเป็นคุณสมบัติของรูปแบบตัวอักษร

3. ผลงานวิจัยของ นายสุรพันธุ์ เอื้อไพบูลย์ เรื่อง "การจดจำตัวอักษรลายมือเขียนภาษาไทยโดยการพิจารณาหัวของตัวอักษร"[12] การวิเคราะห์ตัวอักษรจะพิจารณาจากส่วนหัวของตัวอักษรภาษาไทยที่มีลักษณะเป็นวงกลมเล็ก ในงานวิจัยใช้กล้องเก็บภาพ CCD เป็นอุปกรณ์ข้อมูลเข้า วิธีการวิเคราะห์นี้แบ่งเป็น 4 ขั้นตอน คือ ขั้นแรกเตรียมภาพตัวอักษรไว้ 2 ชุด โดยชุดแรกเป็นชุดปกติและชุดภาพตัวอักษรที่ผ่านการทำให้บาง (Thinning) ขั้นที่สองทำการค้นหาส่วนหัวของตัวอักษรปกติ และหาคุณสมบัติทางโทโพโลยีจากตัวอักษรที่ถูกทำให้บาง ขั้นตอนที่สามทำการแบ่งตัวอักษรออกเป็นกลุ่ม โดยพิจารณาจากจำนวนหัวของตัวอักษร และขั้นตอนสุดท้ายทำการวิเคราะห์ตัวอักษรด้วยเทคนิค 4 แบบ คือ Subhead Region,Feature Code,Head Style และ Width per Height ratio

4. ผลงานวิจัยของ นายชาย เกษมอมรกุล เรื่อง "การออกแบบพจนานุกรมสำหรับการเรียนรู้อักษรคัดลายมือไทย-อังกฤษบนเครื่องไมโครคอมพิวเตอร์"[13] งานวิจัยนี้เป็นการปรับปรุงผลงานวิจัยของ นายประสาน ตังติสานนท์ ด้วยการเพิ่มเติมส่วนการวิเคราะห์ตัวอักษรโดยการค้นหาคุณสมบัติเด่นของตัวอักษรจากพจนานุกรมที่จัดเก็บโดยพิจารณาจากความถี่ของการใช้งานของอักษรแต่ละตัว ทำให้การค้นหารหัสตัวอักษรทำได้รวดเร็วขึ้น

5. ผลงานวิจัยของ นายสุรสิทธิ์ ราตรี เรื่อง "การรู้จำอักษรตัวพิมพ์ภาษาไทยโดยวิธีการค้นหาลักษณะโครงสร้างของลายเส้น"[14] งานวิจัยนี้นำเสนอวิธีการค้นหาลักษณะเด่นของตัวอักษร (Topological feature extraction) ขั้นตอนการวิเคราะห์มี 2 ขั้นตอน ขั้นแรกทำการแบ่งออกเป็น 8 ส่วนในแนวตั้ง,แนวนอน และแนวเส้นทะแยงมุมทั้งสอง ขั้นที่สองพิจารณาคุณสมบัติทางโทโพโลยีของลายเส้นในแต่ละส่วน ซึ่งจะได้รหัสแทนตัวอักษรที่เรียกว่า Q-code

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แนวทางของวิทยานิพนธ์

จากผลงานวิจัยการวิเคราะห์ตัวอักษรภาษาไทยที่ผ่านมามี 2 แนวทางในการพัฒนาคือ วิธีการซ้อนทับ[10] และการหาคุณสมบัติของตัวอักษรทางโทโพโลยี[10,11,12,13,14] วิธีการซ้อนทับที่นำเสนอโดย รศ.ดร.ชม กิ้มปาน ใช้วิธีการสร้างรูปแบบมาตราฐานของตัวอักษรและการวิเคราะห์การกระจายแบบคาร์ฮูแนนโลบ ซึ่งใช้เวลาในการทำประมวลผลนาน ส่วนการหาคุณสมบัติทางโทโพโลยี ที่นำเสนอมานั้นเมื่อนำมาใช้กับตัวอักษรภาษาไทยในกรณีที่ภาพที่สแกนจากเครื่องถ่ายเอกสารมีความเบลอก็อาจทำให้ไม่สามารถวิเคราะห์ได้เนื่องจากคุณสมบัติเด่นของตัวอักษรขาดหายไป อีกทั้งกรณีที่ลักษณะของตัวอักษรมีความคล้ายกันทำให้รหัสของตัวอักษรอาจมีค่าที่เหมือนกันได้

วิทยานิพนธ์ฉบับนี้ขอนำเสนอวิธีการวิเคราะห์ตัวพิมพ์อักษรภาษาไทยวิธีใหม่โดยใช้โครงข่ายประสาทเทียม (Neural network) ซึ่งเป็นวิธีที่ง่ายและมีประสิทธิภาพสูง หลักการของโครงข่ายประสาทเทียมใช้วิธีการเรียนรู้ตัวอย่างข้อมูลเช่นเดียวกับสมองมนุษย์ โดยตัวโครงข่ายจะปรับตัวเองเพื่อสร้างฐานความรู้ให้กับตัวเอง จากนั้นจึงนำความรู้ที่สอนนี้ไปใช้ในการวิเคราะห์ข้อมูล ตัวแบบของโครงข่ายประสาทเทียมที่นำมาประยุกต์ใช้งานวิจัยนี้คือ เคาน์เตอร์พรอพาเกชันเน็ตเวิร์ก (Counterpropagation Network) หรือ CPN ซึ่งตัวแบบนี้ใช้หลักการของการสร้างตารางของตัวอย่างที่นำมาสอนเพื่อนำมาใช้ในการเปรียบเทียบกับข้อมูลที่เข้าสู่โครงข่าย งานวิจัยนี้ได้ดัดแปลงอัลกอริทึม (Algorithm) การสอนโครงข่าย CPN เพื่อช่วยให้สามารถนำมาประยุกต์ใช้งานกับการแยกแยะตัวอักษรภาษาไทย

วัตถุประสงค์ของงานวิจัย

1.เพื่อศึกษาแนวทางการนำโครงข่ายประสาทเทียมมาใช้ในการวิเคราะห์ตัวอักษรภาษาไทย
2.เพื่อศึกษาคุณสมบัติและขั้นตอนการนำของตัวแบบ CPN มาประยุกต์ใช้งาน
3.เพื่อศึกษาปัญหาที่เกิดจากการนำตัวแบบ CPN มาใช้ในการวิเคราะห์ตัวอักษรภาษาไทยและหาแนวทางการแก้ปัญหาที่เกิดขึ้นนี้
4.เพื่อศึกษาประสิทธิภาพในการแยกแยะตัวอักษรภาษาไทยของ CPN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างของวิทยานิพนธ์

โครงสร้างของวิทยานิพนธ์ประกอบด้วย 5 ส่วนคือ บทที่ 2 กล่าวถึงทฤษฎีเบื้องต้นของโครงข่ายประสาทเทียม,บทที่ 3 กล่าวถึงทฤษฎีและปัญหาของ CPN ในการนำมาวิเคราห์ตัวอักษรพิมพ์ภาษาไทย,บทที่ 4 วิธีการแก้ปัญหาของ CPN เพื่อการจดจำตัวพิมพ์อักษรภาษาไทย,บทที่ 5 กล่าวถึงผลการทดลองในการแยกแยะตัวพิมพ์อักษรภาษาไทย และบทที่ 6 สรุปผลและแนวทางการพัฒนาในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีเบื้องต้นของโครงข่ายประสาทเทียม

ปัจจุบันแม้ว่าคอมพิวเตอร์มีความสามารถในการกระมวลผลได้สูงและถูกนำไปใช้ในงานหลายๆด้านได้เป็นอย่างดี ซึ่งเมื่อเทียบกับมนุษย์แล้วคอมพิวเตอร์สามารถทำงานได้เร็วกว่าในงานหลายๆด้าน เช่น การจัดการด้านฐานข้อมูล,การประมวลผลด้าน word processing,โครงข่ายคอมพิวเตอร์ (computer network),งานประมวลด้านการคำนวณ เป็นต้น แต่งานบางอย่างเมื่อนำมาทำด้วยคอมพิวเตอร์และเปรียบเทียบผลกับมนุษย์แล้ว มนุษย์สามารถทำงานได้ดีกว่าและเร็วกว่ามาก เช่น การจดจำหน้าภาพหน้าคน แม้ว่าคนๆนั้นจะเปลี่ยนทรงผมหรือมีสีผิวเปลี่ยนแปลงไปก็ตามมนุษย์เราก็ยังสามารถจดจำคนๆนั้นได้ ซึ่งถ้าเป็นคอมพิวเตอร์แล้วอาจจะไม่สามารถจดจำคนๆนั้นได้อีกทั้งเวลาที่ใช้ในการประมวลผลก็ใช้เวลานานมากด้วย

จากตัวอย่างที่กล่าวข้างต้นมนุษย์เราสามารถทำการวิเคราะห์ได้อย่างรวดเร็ว เนื่องจากสถาปัตยกรรมของสมองมนุษย์มีความแตกต่างจากคอมพิวเตอร์ กล่าวคือ สมองมนุษย์ประกอบด้วยเซลประสาทจำนวนมหาศาลที่เชื่อมต่อกัน ซึ่งเซลเหล่านี้ทำหน้าที่เป็นหน่วยประมวลผลให้กับสมองมนุษย์ การส่งสัญญาณของเซลประสาทระหว่างเซลหนึ่งสู่อีกเซลหนึ่งใช้เวลาเร็วมากคือประมาณสิบส่วนล้านวินาที ซึ่งรูปแบบการส่งสัญญาณทำในลักษณะขนานกัน และวิธีการแก้ปัญหาของสมองมนุษย์ใช้ประสพการณ์ที่เกิดจากการเรียนรู้ในครั้งอดีตนำมาวิเคราะห์เพื่อแก้ปัญหานั้นๆ แต่สถาปัตยกรรมของคอมพิวเตอร์ประกอบด้วยหน่วยประมวลเพียงหน่วยเดียว และการทำงานใช้ชุดคำสั่งสั่งงานเป็นแบบลำดับขั้น อีกทั้งวิธีการแก้ปัญหาต่างๆนั้นจะต้องทราบลำดับขั้นตอนการทำงานที่แน่นนอนของปัญหานั้น

ดังนั้นเมื่อเราต้องการให้คอมพิวเตอร์สามารถจัดการกับปัญหาในลักษณะที่กล่าวข้างต้นจึงได้มีการนำสถาปัตยกรรมของสมองมนุษย์มาเป็นตัวแบบใหม่ในการประมวลผล ซึ่งเรียกว่า Artificial neural systems หรือ โครงข่ายประสาทเทียม (neural networks) [2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงข่ายประสาทเทียมประกอบด้วยหน่วยประมวลผลแบบง่ายๆจำนวนมหาศาลที่เชื่อมต่อกันซึ่งเรียกหน่วยนี้ว่า นิวรอล (neurons),เซล (cell) หรือ โหนด (nodes) และเรียกส่วนที่เชื่อมต่อกันระหว่างโหนดนี้ว่า อินเตอร์คอนเน็กชัน (interconnection) โดยที่อินเตอร์คอนเน็กชันจะเก็บความรู้ที่ใช้ในการแก้ปัญหานั้นๆของโครงข่าย ซึ่งความรู้ที่ใช้ในการแก้ปัญหานี้ได้มาจากการเรียนรู้ตัวอย่างของปัญหานั้น เราสามารถเปรียบเทียบโครงข่ายประสาทเทียมกับสมองมนุษย์ได้ในแง่ของการเก็บข้อมูลต่างๆในรูปของแพตเทิร์นของอินเตอร์คอนเน็กชัน (Pattern of Interconnections) และลักษณะการแก้ปัญหาด้วยวิธีการเรียนตัวอย่างซึ่งเทียบได้กับการสะสมประสพการณ์ของมนุษย์

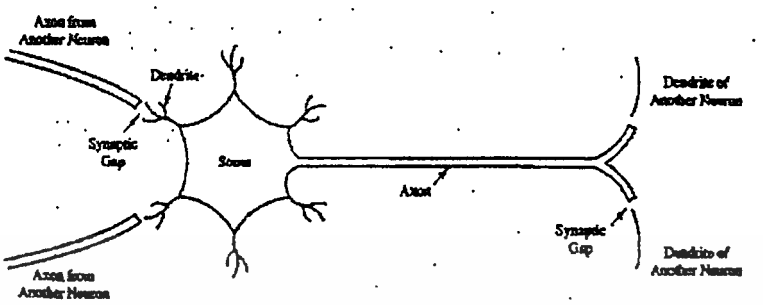
โครงข่ายประสาทเทียมได้ถูกนำมาประยุกต์ในงานด้านต่างๆหลายด้าน เช่น storing และ recall data,pattern classification,การแบ่งกลุ่มแพตเทิร์นที่มีลักษณะเหมือนกัน เป็นต้น[3] โครงข่ายประสาทเทียมมีคุณสมบัติซึ่งเรียกว่า generalization กล่าวคือ โครงข่ายสามารถวิเคราะห์หรือสร้างคำตอบสำหรับปัญหาซึ่งมีลักษณะที่เหมือนหรือคล้ายกับตัวอย่างที่นำมาสอนได้อย่างมีประสิทธิภาพ โดยที่ปัญหานั้นไม่เคยถูกนำมาสอนให้กับโครงข่ายเลย

ตัวแบบของเซลประสาทเทียม (Artificial Neuron Model)

ลักษณะเซลประสาททางชีวภาพประกอบด้วย 3 ส่วนหลัก คือ เดนไดรต์ (dendrites),โซมา (soma) หรือตัวเซล (cell body) และแอคซอน (axon) ส่วนที่เชื่อมระหว่างตัวเซลกับแอคซอนของเซลอื่นๆที่อยู่รอบๆข้างเรียกว่า ไซแนปส์ (synapse) โดยตัวเดนไดรต์จะรับสัญญาณจากเซลประสาทที่อยู่รอบๆข้างผ่านทางไซแนปส์ด้วยปฏิกิริยาทางเคมี ซึ่งสัญญาณที่เข้ามานี้คือ ประจุอิเล็กตรอนโดยปฏิกิริยาเคมีที่เกิดขึ้นจะทำการปรับเปลี่ยนความถี่ของสัญญาณที่เข้ามา ตัวเซลทำหน้าที่รวมสัญญาณที่เข้ามาแล้วทำการส่งสัญญาณออกให้กับแอคซอนเพื่อผ่านต่อให้กับเซลอื่นๆ ซึ่งสัญญาณที่ออกจากตัวเซลมีลักษณะเป็นแบบสัญญาณกระตุ้นให้กับเซลอื่น กล่าวคือ ถ้าค่าสัญญาณที่ส่งออกจากตัวเซลมีจำนวน 100 ครั้งต่อวินาทีถือว่าค่าสัญญาณที่ส่งออกเป็นสถานะการกระตุ้น (fire) โดยทั่วไปเราสนใจสัญญาณที่ออกจากเซลประสาทในรูปของสถานะการกระตุ้นและไม่กระตุ้น (not fire) ให้กับเซลที่อยู่รอบข้างมากกว่าสนใจเงื่อนไขที่เกี่ยวข้องที่ทำให้เซลประสาทเกิดสถานะนี้ขึ้นมา ภาพที่ 2 แสดงลักษณะของเซลประสาททางชีววิทยา[3,8]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

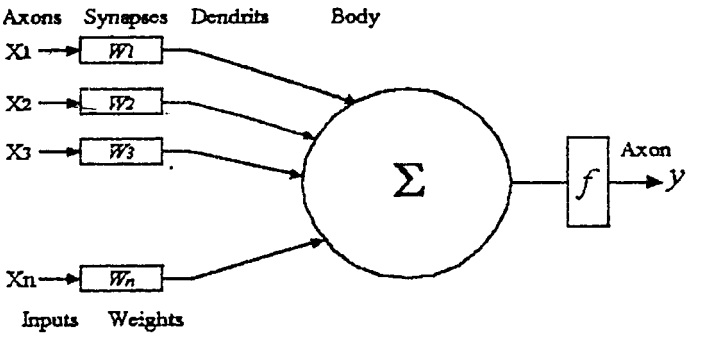
ภาพที่ 2



แสดงลักษณะของเซลประสาท

ตัวแบบเซลประสาทเทียมที่ใช้ใน artificial neural network มีคุณลักษณะพื้นฐานเหมือนกับเซลประสาททางชีววิทยา ภาพที่ 3 แสดงลักษณะของตัวแบบเซลประสาทเทียมที่ใช้กันโดยทั่วไปซึ่งใช้พื้นฐานจากตัวแบบที่เสนอโดย McCulloch และ Pitt ในปี 1943 จากภาพที่ 3 เซลประสาทเทียมรับสัญญาณ x_1 - x_n จากเซลรอบข้าง โดยไซแนปส์หรืออินเตอร์คอนเน็กชันจะทำการปรับเปลี่ยนหรือถ่วงน้ำหนักค่าสัญญาณที่เข้ามาด้วย w_1 - w_n สมการที่ 2.1 แสดงการหาค่าผลรวมสัญญาณที่เข้าสู่เซลประสาทเทียมโหนดที่ i ซึ่งเรียกว่าค่า net input และ y คือค่าสัญญาณที่ออกจากเซลประสาทเทียมซึ่งจะปรากฏที่แอคซอน โดยค่าสัญญาณนี้จะส่งผ่านทางฟังก์ชันกระตุ้น (activation function) $f(\ )$ ซึ่งฟังก์ชันนี้ทำหน้าที่เลือกผ่านค่าสัญญาณให้กับแอคซอน โดยจะทำการย่อขนาดของผลรวมสัญญาณที่ออกจากตัวเซลให้อยู่ในช่วง 0 ถึง 1 ในบางครั้งอาจไม่มีการใช้ฟังก์ชันกระตุ้นก็ได้ สมการที่ 2.2 แสดงการหาค่าฟังก์ชันกระตุ้นของสมการที่ 2.1[4]

ภาพที่ 3



แสดงตัวแบบของเซลประสาทเทียม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\text{net}_i = \sum_j^n x_j w_{ij} \qquad \text{.....(2.1)}$$

$$y_i = f_i(\text{net}_i) \qquad , i = 1,2,....,m \text{ .....(2.2)}$$

โดยที่ x_j : ค่าสัญญาณลำดับที่ j ที่เข้าสู่เซลประสาทเทียมโหนดที่ i

w_{ij} : ค่าถ่วงน้ำหนักสัญญาณลำดับที่ j ที่เข้าสู่เซลประสาทเทียมโหนดที่ i

net_i : ค่า net input ของเซลประสาทเทียมโหนดที่ i

$f_i(\,)$: ฟังก์ชันกระตุ้นของเซลประสาทเทียมโหนดที่ i

y_i : ค่าสัญญาณที่ออกจากเซลประสาทเทียมโหนดที่ i

m : จำนวนเซลประสาทเทียมของโครงข่าย

ค่าของอินเตอร์คอนเน็กชันของเซลประสาทเทียมอาจมีค่าเป็นบวก,ลบ หรือศูนย์ก็ได้ ดังนั้นในการปรับแต่งค่าสัญญาณที่เข้าสู่เซลประสาทด้วยค่าถ่วงน้ำหนักของอินเตอร์คอนเน็กชัน ถ้าข้อมูลที่เข้าสู่เซลประสาทเทียมมีค่าเป็นบวกและค่าของอินเตอร์คอนเน็กชันเป็นบวกด้วยก็จะก่อให้เกิดสภาพเสริมกัน (Excitatory) และถ้าค่าของอินเตอร์คอนเน็กชันมีค่าเป็นลบก็จะลดค่าความเข้มของข้อมูลที่เข้ามานั้น ส่วนในกรณีที่ค่าของอินเตอร์คอนเน็กชันเท่ากับศูนย์ อินเตอร์คอนเน็กชันตรงส่วนนั้นหยุดทำงาน

ค่าผลลัพธ์ที่ได้จากฟังก์ชันกระตุ้นนี้ทำให้เซลประสาทเทียมมีลักษณะเหมือนกับเซลประสาททางชีววิทยา กล่าวคือค่าผลลัพธ์ที่ได้จากฟังก์ชันกระตุ้นเป็นค่าที่อยู่ในช่วง 0 ถึง 1 และค่าสัญญาณที่ออกจากเซลประสาททางชีววิทยามีลักษณะเป็นสถานะของการกระตุ้น ดังนั้นเราสามารถแทนความหมายสถานะการกระตุ้นของเซลประสาทด้วยค่า 1 และการไม่กระตุ้นด้วย 0 ได้ รูปแบบฟังก์ชันกระตุ้นที่นิยมใช้มีดังต่อไปนี้[8,9]

1. Step Function ฟังก์ชันนี้นิยมใช้ในโครงข่ายแบบชั้นเดียว (Single Layer) ซึ่งใช้ในการแปลงค่า net input ที่อยู่ในรูปของค่าต่อเนื่องให้อยู่ในรูปของไบนารี (binary) คือ 1 และ 0 , หรือไบโพลล่า (bipolar) คือ 1 และ -1 ฟังก์ชันนี้จะใช้ค่าเทรส โฮลด์ (threshold) θ ในการกำหนดการแปลงค่า ซึ่งเรียกฟังก์ชันนี้ว่า Threshold function หรือ Heavisde function ตัวอย่างโครงข่ายที่ใช้ฟังก์ชันนี้ได้แก่ Adaline สมการที่ 2.3 แสดงสมการของ Threshold function

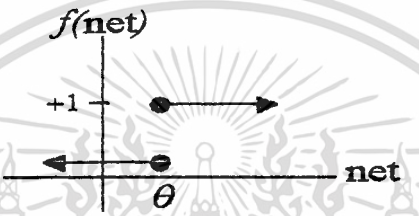
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$f(\text{net}) = \begin{cases} 1 \text{ if net} \geq \theta \\ 0 \text{ if net} < \theta \end{cases} \quad \text{.....(2.3)}$$

โดยที่ net : ค่า net input ที่ของเซลประสาทเทียม

θ : ค่าเทรสโฮลด์ที่ใช้ในการแปลงค่าของเซลประสาทเทียม

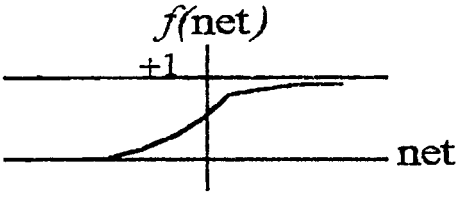
ภาพที่ 4



แสดงลักษณะของ Binary Step โดยใช้ค่า Threshold

2. Sigmoid function (S-shaped curves) ฟังก์ชันนี้ปกติแล้วนิยมนำมาใช้ในการแปลงค่า net input ให้อยู่ในรูปของช่วง 0 ถึง 1 ซึ่งเรียกว่า Binary sigmoid ฟังก์ชัน หรืออยู่ในช่วง -1 ถึง 1 ซึ่งเรียกว่า Bipolar sigmoid ฟังก์ชันที่นำมาใช้ในการแปลงค่าเป็น Sigmoid function โดยทั่วไปแล้วใช้ logistic function และ hyperbolic tangent function แสดงในสมการที่ 2.4 และ 2.5 ตามลำดับ ซึ่งฟังก์ชันเหล่านี้มีประโยชน์อย่างมากในการสอนโครงข่ายแบบแพร่ย้อนกลับ (Backpropagation network) ภาพที่ 5 และ 6 แสดงตัวอย่างของ Binary sigmoid และBipolar sigmoid ด้วย logistic function และ hyperbolic tangent function ตามลำดับ

ภาพที่ 5

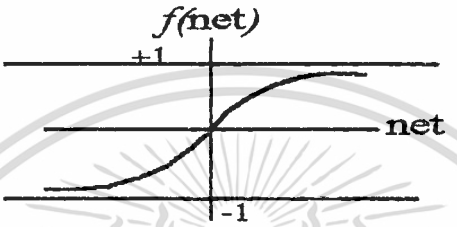


แสดงลักษณะของ Binary sigmoid

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$f(\text{net}) = 1 / (1+e^{-\text{net}}) \quad \text{.....(2.4)}$$

ภาพที่ 6



แสดงลักษณะของ Bipolar sigmoid

$$f(\text{net}) = (e^{\text{net}}-e^{-\text{net}})/(e^{\text{net}}+e^{-\text{net}}) \quad \text{.....(2.5)}$$

โดยที่ net : ค่า net input ที่ของเซลประสาทเทียม

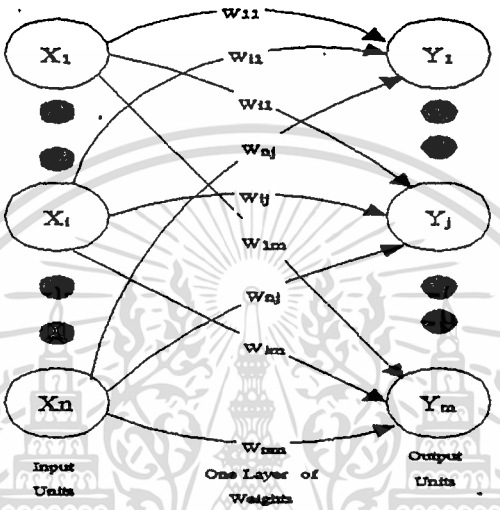
<u>สถาปัตยกรรมของโครงข่ายประสาทเทียม</u> (Neural network Architecture)

โครงข่ายประสาทเทียมประกอบด้วยเซลประสาทเทียมหรือโหนดจำนวนมากที่เชื่อมต่อกัน ซึ่งการเชื่อมต่อจะแบ่งออกเป็นกลุ่มย่อยเรียกว่า ชั้น (Layer) โดยทั่วไปแล้วแบ่งออกเป็น 2 แบบ[3] คือ โครงข่ายแบบชั้นเดียว (Single-layer) และโครงข่ายแบบหลายชั้น (Multilayer) การกำหนดจำนวนชั้นของโครงข่ายนี้ส่วนที่เป็นหน่วยรับข้อมูลเข้า (input unit) จะไม่ถูกนับด้วยเนื่องจากเป็นส่วนที่ไม่มีการคำนวณ (หนังสือบางเล่มถือว่าหน่วยข้อมูลเข้านี้เป็นชั้นของโครงข่ายด้วย) ดังนั้นเราอาจกล่าวได้ว่าจำนวนชั้นของโครงข่ายคือ จำนวนชั้นของอินเตอร์คอนเน็กชันที่ถูกถ่วงน้ำหนัก โดยปกติแล้วโครงข่ายแบบหลายชั้นสามารถแก้ปัญหาที่มีความซับซ้อนได้ดีกว่าโครงข่ายแบบชั้นเดียว และในบางปัญหานั้นไม่สามารถแก้ปัญหาด้วยโครงข่ายแบบชั้นเดียวได้ ภาพที่ 7 แสดงลักษณะการต่อของโครงข่ายชั้นเดียว ซึ่งตัวแบบที่สถาปัตยกรรมแบบนี้ ได้แก่ BAM ,

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

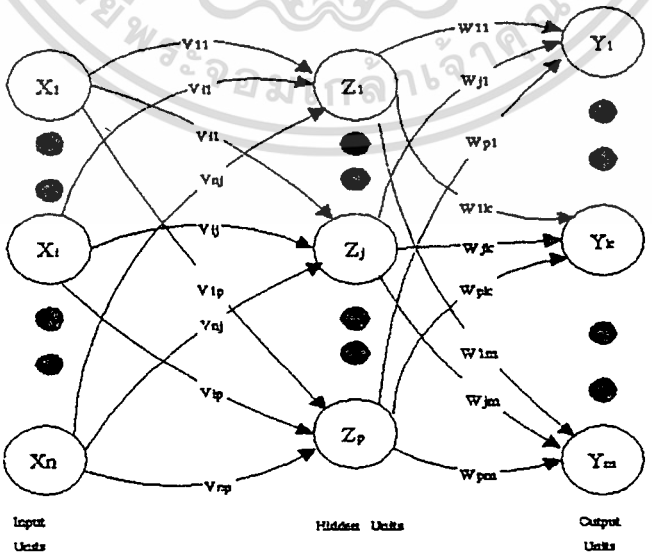
Hopfield ส่วนภาพที่ 8 แสดงสถาปัตยกรรมแบบหลายชั้นซึ่งตัวแบบได้แก่ Backpropaga-tion,Self-Organizing Maps,Couterpropagation

ภาพที่ 7



แสดงตัวอย่างของโครงข่ายแบบชั้นเดียว

ภาพที่ 8



แสดงตัวอย่างของโครงข่ายแบบหลายชั้น

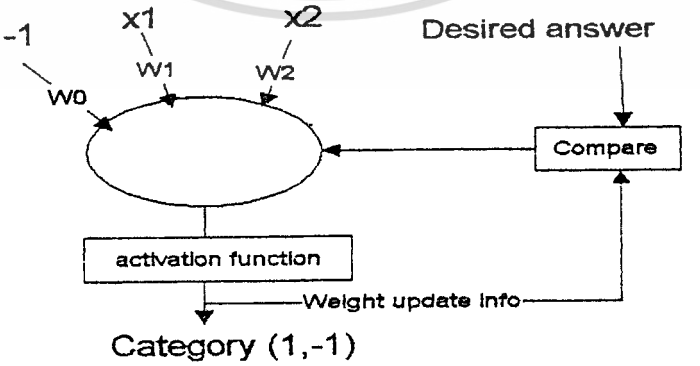
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การฝึกสอนโครงข่ายประสาทเทียม

จากที่กล่าวข้างต้นก่อนการนำโครงข่ายประสาทเทียมไปใช้งานจะต้องมีการสร้างความรู้ (knowledge) ให้กับโครงข่ายก่อนซึ่งเรียกขั้นตอนนี้ว่า การฝึกสอนโครงข่าย ข้อมูลที่นำมาเป็นชุดฝึกสอนให้แก่โครงข่ายคือตัวอย่างข้อมูลของปัญหาที่ต้องการนำโครงข่ายไปใช้ในการแก้ปัญหา โดยการเรียนรู้ของโครงข่ายจะทำการปรับค่าถ่วงน้ำหนักของอินเตอร์คอนเน็กชันที่เชื่อมต่อกันในแต่ละชั้นของโครงข่ายเพื่อสร้างฐานความรู้ของปัญหานั้นๆให้กับโครงข่าย ลักษณะการสอนของโครงข่ายประสาทเทียมแบ่งได้เป็น 2 แบบ คือ การฝึกสอนแบบมีผู้สอน (supervised training) และการฝึกแบบไม่มีผู้สอน (unsupervised training)

การฝึกสอนแบบมีผู้สอน (Supervised training) ข้อมูลที่นำเข้าสู่โครงข่ายด้วยการสอนวิธีนี้ประกอบด้วย 2 ส่วน[5] คือ อินพุตเวคเตอร์ (input vector) ที่เป็นตัวอย่างข้อมูลของปัญหานั้นๆ และเวคเตอร์เป้าหมาย (target vector) ซึ่งเป็นผลลัพธ์ที่เราต้องการให้โครงข่ายสร้างเมื่อมีการนำอินพุตเวคเตอร์ของตัวอย่างข้อมูลมาสอนให้กับโครงข่าย ขณะสอนโครงข่ายด้วยวิธีนี้จะมีการกำหนดค่าผลลัพธ์เป้าหมายให้กับแต่ละอินพุตเวคเตอร์ที่นำเข้ามาสอน โดยโครงข่ายจะนำค่าผิดพลาดที่ได้จากการคำนวณของโครงข่าย (actual output) กับค่าผลลัพธ์เป้าหมายที่กำหนดขึ้นมาใช้ในการเรียนหรือปรับค่าถ่วงน้ำหนักของอินเตอร์คอนเน็กชัน ตัวแบบโครงข่ายที่การเรียนรู้ลักษณะนี้ได้แก่ Adaline,Perceptron, Back-propagation เป็นต้น

ภาพที่ 9



แสดงตัวอย่างการปรับค่าถ่วงน้ำหนักของเพอร์เซปตรอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 9 แสดงตัวอย่างโครงสร้างการสอนของเพอร์เซปตรอน (Perceptron) เพื่อใช้ในแยกข้อมูลสองกลุ่ม ตารางที่ 1 แสดงข้อมูลที่เข้าสู่โครงข่ายและผลลัพธ์ของข้อมูลนั้นที่นำมาสอนโครงข่าย โดยที่ x_1, x_2 เป็นข้อมูลที่เข้าสู่โครงข่าย และ F คือผลลัพธ์ที่ต้องการ จากภาพที่ 9 ตัวเซลของเพอร์เซปตรอนรับข้อมูลเข้า 3 ทางคือ -1, x_1, x_2 โดยเรียกข้อมูลเข้าที่เป็นค่าคงที่ -1 (อาจเป็น +1) ว่า ไบเอส (bias)

ตารางที่ 1

ข้อมูลที่เข้าสู่โครงข่าย		ผลลัพธ์ที่ต้องการ
x_2	x_2	F
-1	-1	-1
-1	1	-1
1	-1	-1
1	1	1

แสดงข้อมูลที่นำมาสอนเพอร์เซปตรอนและผลลัพธ์ที่ต้องการ

สมการที่ 2.6 และ 2.7 แสดงการหาค่าของฟังก์ชันการกระตุ้นและการปรับค่าถ่วงน้ำหนักของเพอร์เซปตรอนตามลำดับ

$$y = +1 \quad \text{ถ้า net} \geq 0$$
$$= -1 \quad \text{ถ้า net} < 0 \quad \text{.....(2.6)}$$

$$\mathbf{W}(t+1) = \mathbf{W}(t) + \alpha(\mathbf{EX}) \quad \text{.....(2.7)}$$

โดยที่ $\mathbf{E}$ = ค่าผิดพลาดที่ใช้ปรับค่าถ่วงน้ำหนัก (desired category-actual category)

α = ค่า learning rate ($0 < \alpha < 1$)

$\mathbf{W}$ = ค่าเวคเตอร์ของค่าถ่วงน้ำหนักของเซลเพอร์เซปตรอนที่เวลา t

$\mathbf{X}$ = ค่าอินพุตเวคเตอร์ (input vector) ที่เข้าสู่เซลเพอร์เซปตรอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>ตัวอย่างการสอนเซลประสาทเพอร์เซปตรอน</u>

ขั้นตอนข้างล่างนี้แสดงตัวอย่างการสอนเพอร์เซปตรอนเพื่อให้สามารถแยกแยะข้อมูลของตารางที่ 1 บางส่วน โดยกำหนดให้ค่า $w_0 = 0$, $w_1 = 0$, $w_2 = 0$ และให้ค่า α มีค่าเท่ากับ 0.5

1. $x_1 = 1$,$x_2 = 1$ และ F = 1

net = (0)(-1) + (0)(1) + (0)(1) = 0

y = 1 (เนื่องจาก net > 0)

2. $x_1 = 1$,$x_2 = -1$ และ F = -1

net = (0)(-1) + (0)(1) + (0)(-1) = 0

y = 1 (เนื่องจาก net > 0)

ปรับค่าถ่วงน้ำหนัก

E = -1 - 1 = -2

$w_0 = 0 + 0.5(-2)(-1) = 1$

$w_1 = 0 + 0.5(-2)(1) = -1$

$w_2 = 0 + 0.5(-2)(-1) = 1$

3. $x_1 = -1$,$x_2 = 1$ และ F = -1

net = (1)(-1) + (-1)(-1) + (1)(1) = 1

y = 1 (เนื่องจาก net > 0)

ปรับค่าถ่วงน้ำหนัก

E = -1 - 1 = -2

$w_0 = 1 + 0.5(-2)(-1) = 2$

$w_1 = -1 + 0.5(-2)(-1) = 0$

$w_2 = 1 + 0.5(-2)(1) = 0$

4. $x_1 = -1$,$x_2 = -1$ และ F = -1

net = (2)(-1) + (0)(-1) + (0)(-1) = -2

y = -1 (เนื่องจาก net < 0)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. $x_1 = 1, x_0 = 1$ และ $\mathbf{F} = 1$

net = (2)(-1) + (0)(1) + (0)(1) = -2

y = -1 (เนื่องจาก net < 0)

ปรับค่าถ่วงน้ำหนัก

$\mathbf{E}$ = 1 - (-1) = 2

w_0 = 2 + 0.5(2)(-1) = 1

w_1 = 0 + 0.5(2)(1) = 1

w_2 = 0 + 0.5(2)(1) = 1

ตารางที่ 2 แสดงผลสรุปการสอนเพอร์เซปตรอน ซึ่งการสอนมีจำนวน 3 รอบ โดยครั้งที่ 1-4,5-8 และ 9 เป็นการสอนรอบที่ 1,2 และ 3 ตามลำดับ จากตารางในรอบที่ 2 และ 3 พบว่าค่าถ่วงน้ำหนักไม่มีการเปลี่ยนแปลงเลยและโครงข่ายสามารถแยกแยะข้อมูลได้ถูกต้อง ซึ่งแสดงให้เห็นว่าค่าถ่วงน้ำหนักของเซลประสาทเทียมเข้าสู่จุดสมดุลแล้วจึงทำการหยุดสอน

ตารางที่ 2

ครั้ง	Input		net	actual	desire	Error	New	New	New
	x_1	x_2		category	category	(E)	w_0	w_1	w_2
0	-	-	-	-	-	-	0	0	0
1	1	1	0	1	1	0	0	0	0
2	1	-1	0	1	-1	-2	1	-1	1
3	-1	1	1	1	-1	-2	2	0	0
4	-1	-1	-2	-1	-1	0	2	0	0
5	1	1	-2	-1	1	-2	1	1	1
6	1	-1	-1	-1	-1	0	1	1	1
7	1	1	-1	-1	-1	0	1	1	1
8	-1	-1	-3	-1	-1	0	1	1	1
9	1	1	1	1	1	0	1		

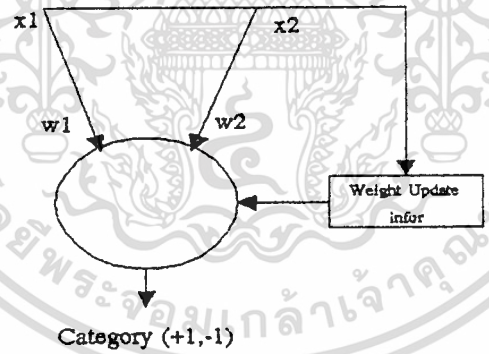
แสดงขั้นตอนของเพอร์เซปตรอนจนสามารถแยกข้อมูลที่นำมาสอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การฝึกสอนแบบไม่มีผู้สอน (Unsupervised training) ข้อมูลที่นำมาใช้ในการสอนโครงข่ายด้วยวิธีนี้มีเพียงอินพุตเวคเตอร์ที่เป็นตัวอย่างข้อมูลของปัญหานั้นๆเพียงอย่างเดียว โดยขณะสอนจะไม่มีการกำหนดค่าผลลัพธ์เป้าหมายที่ต้องการเพื่อใช้ในการเรียนหรือปรับค่าถ่วงน้ำหนักให้กับโครงข่าย ซึ่งลักษณะการปรับค่าถ่วงน้ำหนักของโครงข่ายจะใช้ข้อมูลที่นำมาสอนในการปรับเปลี่ยนค่า โดยกลุ่มของโหนดที่ถูกปรับค่าถ่วงน้ำหนักนี้จะเป็นกลุ่มเดียวกับกลุ่มข้อมูลที่ให้ค่าผลลัพธ์คล้ายๆกัน การฝึกสอนด้วยวิธีนี้ใช้เวลาในการเรียนตัวอย่างได้รวดเร็วกว่าการฝึกสอนแบบมีผู้สอน ตัวอย่างตัวแบบที่ใช้การเรียนลักษณะนี้ เช่น Kohonen self-organizon Maps (SOM), Counterpropagation network (CPN), Adaptive Resonance Theory (ART) ภาพที่ 10 แสดงรูปแบบการสอนแบบของโคโฮแนน (Kohonen) และสมการที่ 2.8 เป็นสมการที่ในการปรับค่าถ่วงน้ำหนักของโคโฮแนน ซึ่งจากสมการจะสังเกตุได้ว่าทิศทางของค่าถ่วงน้ำหนักปรับตามอินพุตเวคเตอร์ที่เข้าสู่โครงข่าย[6]

ภาพที่ 10.



แสดงตัวอย่างการปรับค่าถ่วงน้ำหนักของโคโฮแนน

$$\mathbf{W}(t+1) = \mathbf{W}(t) + \alpha(\mathbf{X} - \mathbf{W}(t)) \quad \text{.....(2.8)}$$

โดยที่ α = ค่า learning rate ($0 < \alpha < 1$)

$\mathbf{W}$ = ค่าเวคเตอร์ของค่าถ่วงน้ำหนัก (weight vector) ที่เวลา t

$\mathbf{X}$ = ค่าอินพุตเวคเตอร์ (input vector) ที่เข้าสู่โครงข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

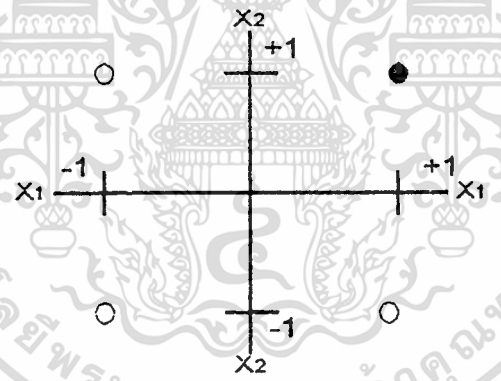
27112

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

ปกติทั่วไปแล้วผลลัพธ์ที่ออกจากเซลประสาทเทียมเราต้องการคำตอบในรูปของตรรกศาสตร์ คือ ใช่หรือไม่ใช่ กล่าวคือคำตอบใช่เมื่อข้อมูลที่เข้าสู่เซลประสาทเทียมเป็นสมาชิกของกลุ่มนั้น และไม่ใช่เมื่อข้อมูลนั้นไม่ได้เป็นสมาชิก ซึ่งเทียบกับผลลัพธ์ที่ได้จากเซลประสาทเทียมคือ 1 และ -1 (0) การที่เราต้องการให้โครงข่ายสร้างคำตอบในลักษณะนี้ต้องใช้ฟังก์ชันกระตุ้นแบบ step function เนื่องจากผลลัพธ์ที่ออกจากฟังก์ชันนี้มีสองสถานะเช่นกัน

จากตัวอย่างการสอนเพอร์เซปตรอนข้างต้นเราสามารถแสดงความสัมพันธ์ของกลุ่มข้อมูลที่นำมาฝึกสอนด้วยกราฟภาพที่ 11 โดยจุดดำแสดงเวกเตอร์ (1,1) ซึ่งให้ผลลัพธ์คือ 1 และจุดขาวแสดงเวกเตอร์ (1,-1),(-1,1) และ (-1,-1) ซึ่งให้ผลลัพธ์ -1

ภาพที่ 11



แสดงความสัมพันธ์ของข้อมูลเชิงเรขาคณิตของตัวอย่างเพอร์เซปตรอน

ซึ่งจากตัวอย่างนี้สามารถเขียนสมการหาค่า net input ได้ดังสมการที่ 2.9

$$\text{net} = \sum_{i=1}^{n} x_i w_i - w_0 \quad \text{เมื่อ } x_0 \text{ เท่ากับ } -1 \quad \text{.....(2.9)}$$

โดยที่

$$f(\text{net}) = +1 \quad \text{ถ้า net} \geq 0$$
$$= \quad -1 \quad \text{ถ้า net} < 0$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเรียกขอบเขตระหว่างพื้นที่ $f(\text{net}) > 0$ และ $f(\text{net}) < 0$ ว่าขอบเขตการตัดสินใจ (decision boundary) ซึ่งเราสามารถหาเส้นแบ่งได้จากสมการที่ 2.10

$$\sum_{i=1}^{n} x_i w_i - w_0 = 0 \quad \text{.....(2.10)}$$

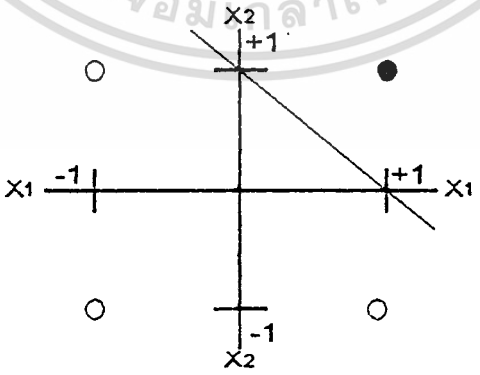
ซึ่งสมการที่ 2.10 เป็นสมการเส้นตรง (line),เพลน (plane) หรือ ไฮเปอร์เพลน (hyperplane)

เนื่องจากข้อมูลที่เข้าสู่เพอร์เซปตรอนมีลักษณะเป็นคู่คือ (x_1, x_2) และค่าของน้ำหนักที่ได้จากการสอนครั้งสุดท้ายคือ $w_0 = 1$, $w_1 = 1$ และ $w_0 = 1$ เมื่อนำมาแทนค่าในสมการของเส้นขอบเขตการตัดสินใจแสดงได้ดังสมการที่ 2.11 นี้

$$x_1 + x_2 = 1 \quad \text{.....(2.11)}$$

จากการแทนค่าสมการของเส้นแบ่งขอบเขตการตัดสินใจข้างบนเมื่อนำวาดเส้นตรงแสดงดังภาพที่ 12 จากภาพที่ 12 จะเห็นว่าเส้นตรงนั้นสามารถแบ่งขอบเขตระหว่างข้อมูลทั้งสองกลุ่มได้ ซึ่งเราเรียกลักษณะแบ่งแบบนี้ว่า การแบ่งแบบเชิงเส้น (Linear separability)

ภาพที่ 12

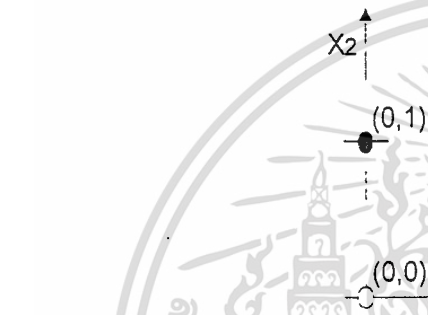


กราฟเส้นแบ่งขอบเขตที่ได้จากสอนเพอร์เซอร์ตรอน

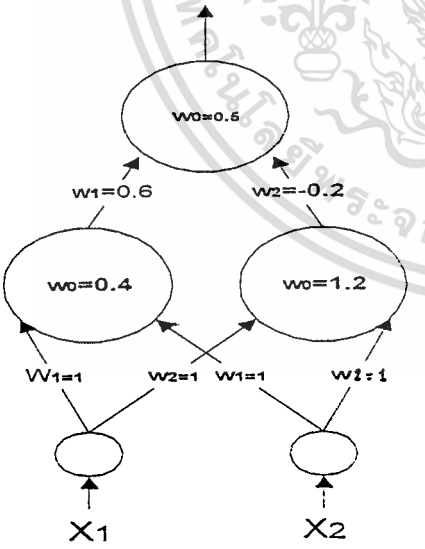
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตัวอย่างปัญหาของเพอร์เซปตรอนข้างต้นเป็นการทำโอเปอร์เรชัน (operation) AND ซึ่งมีลักษณะเป็นแบบเชิงเส้น แต่กรณีที่ลักษณะของปัญหาเป็นแบบไม่เป็นเชิงเส้น (nonlinear) เช่น XOR แสดงดังกราฟภาพที่ 13.1 ซึ่งเพอร์เซปตรอนไม่สามารถแก้ปัญหานี้ได้ ดังนั้นเพื่อให้สามารถแก้ปัญหานี้ได้จึงมีการเพิ่มชั้นของโครงข่ายแสดงดังภาพที่ 13.2 ภาพที่ 13.3 แสดงเส้นแบ่งที่ได้จากการสอนเพอร์เซปตรอนที่เพิ่มชั้นเพื่อใช้แยกข้อมูลปัญหา XOR[12]

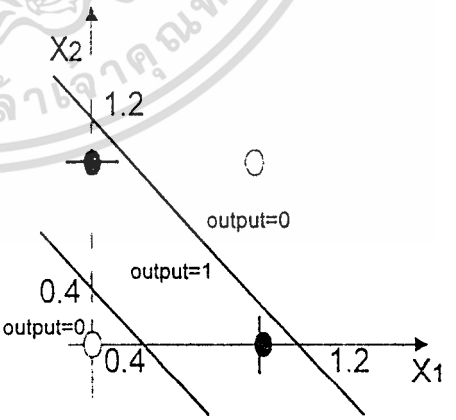
ภาพที่ 13



13.1) แสดงกราฟของปัญหา XOR



13.2) แสดงการเพิ่มชั้นของเพอร์เซปตรอน



13.3) แสดงเส้นแบ่งที่ได้จากสอนภาพที่ 13.1

แสดงการแก้ปัญหาของ XOR ด้วยเพอร์เซปตรอนที่เพิ่มชั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

ทฤษฎีและปัญหาของ CPN ที่ใช้ในการวิเคราะห์ตัวอักษรภาษาไทย

ตัวแบบโครงข่ายประสาทเทียมของเคาน์เตอร์พรอพาเกชันหรือ CPN ถูกพัฒนาขึ้นโดย Hecht Nielsen ตัวแบบนี้มีโครงสร้างเป็นแบบหลายชั้นซึ่งเกิดจากการรวมกันของชั้นโคโฮแนน (Kohonen layer) หรือชั้นคอมเพททิทิพ (Competitive layer) และชั้นผลลัพธ์ (output layer) ที่มีการเรียนรู้แบบ Grossberg learning[7] โครงข่าย CPN ได้ถูกนำไปประยุกต์ใช้งานประเภทต่างๆ เช่น การบีบอัดข้อมูล (data compressing), การประมาณค่าฟังก์ชัน (approximate function)

การเรียนของโครงข่าย CPN จะทำการปรับตัวเองในลักษณะการสร้างตาราง look-up table ของตัวอย่างที่นำมาสอนเพื่อใช้ประมาณค่าข้อมูลที่เข้าสู่โครงข่าย รูปแบบการฝึกสอนของ CPN มีลักษณะเป็นแบบการฝึกแบบไม่มีผู้สอน โดยแบ่งการสอนออกเป็น 2 ขั้นตอน ขั้นตอนแรกเป็นการสอนที่ชั้นคอมเพททิทิพเพื่อทำการแบ่งกลุ่มข้อมูลที่มีลักษณะเหมือนกันออกเป็นกลุ่มๆ (cluster) และขั้นตอนที่สองเป็นการสอนที่ชั้นผลลัพธ์ซึ่งการสอนในชั้นนี้เพื่อใช้บอกหรือระบุชื่อกลุ่มๆของข้อมูลที่ถูกแบ่งจากชั้นของคอมเพททิทิพ

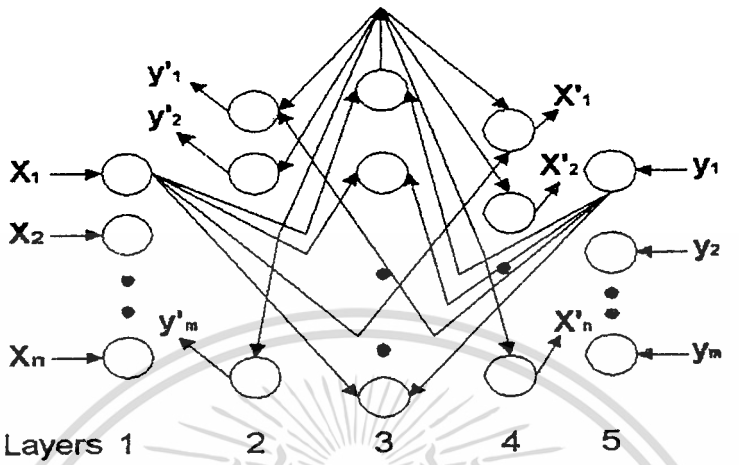
พระจอมเกล้าเจ้าคุณทหาร

<u>โครงสร้างของโครงข่ายเคาน์เตอร์พรอพาเกชัน</u>

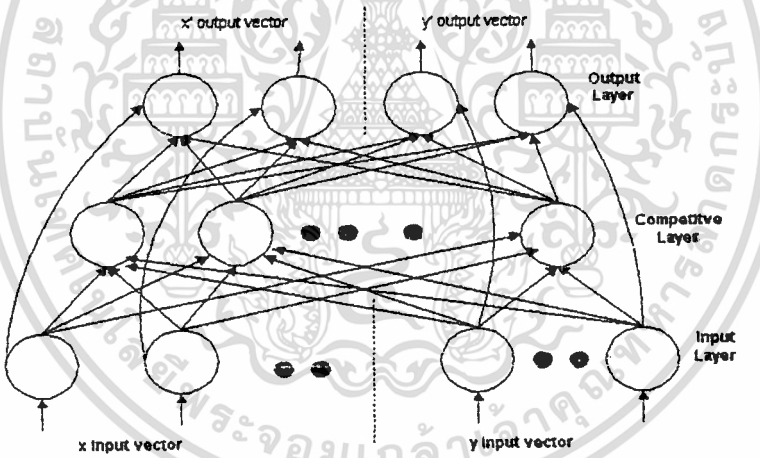
โครงสร้างของ CPN แบ่งออกเป็น 2 ประเภทด้วยกันคือ เคาน์เตอร์พรอพาเกชันแบบเต็มรูปแบบ (full counterpropagation) และเคาน์เตอร์พรอพาเกชันแบบแพร่ไปข้างหน้า (forward counterpropagation) ภาพที่ 14.1 แสดงสถาปัตยกรรมของเคาน์เตอร์พรอพาเกชันแบบเต็มรูปแบบ ซึ่งประกอบด้วย 5 ชั้น จากภาพชั้นที่ 1 และ 5 เป็นชั้นข้อมูลเข้า (input layer),ชั้นของผลลัพธ์ (output layer) คือชั้นที่ 2 และ 4 ,และชั้นที่ 3 คือชั้นคอมเพททิทิพ (Competitive layer) ซึ่งโครงข่ายนี้ถูกออกแบบมาเพื่อใช้ประมาณค่าฟังก์ชันแบบต่อเนื่อง (continuous function) $y=\Phi(x)$ และฟังก์ชันแบบอินเวอร์ส (inverse function) $x=\Phi^{-1}(y)$ โดยที่ Φ เป็นฟังก์ชันแบบต่อเนื่อง[2]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 14



14.1) แสดงสถาปัตยกรรมของเคาน์เตอร์พรอพาเกชันแบบเต็มรูปแบบ (Original)



14.2) แสดงสถาปัตยกรรมของเคาน์เตอร์พรอพาเกชันแบบเต็มรูปแบบที่เขียนใหม่

แสดงสถาปัตยกรรมของเคาน์เตอร์พรอพาเกชันแบบเต็มรูปแบบ

ภาพที่ 14.2 แสดงการเขียนภาพที่ 14.1 ใหม่ จากภาพอินพุตเวคเตอร์ที่เข้าสู่โครงข่าย CPN แบบเต็มรูปแบบแบ่งออกเป็น 2 ส่วนคือ อินพุตเวคเตอร์ X และ Y ซึ่งอินพุตเวคเตอร์ทั้งสองนี้ถูกใช้ประมาณค่าฟังก์ชัน $y=\Phi(x)$ และ $x=\Phi^{-1}(y)$ ตามลำดับ

ลักษณะการเชื่อมต่อของโครงข่าย CPN แบบเต็มรูปแบบแสดงได้ดังนี้คือ เส้นอินเตอร์คอนเน็กชันที่เชื่อมระหว่างชั้นข้อมูลเข้า (input layer) กับชั้นคอมเพททิทิพ (competitive layer) นั้นทุกเส้นต่อแบบทั่วถึงกันหมด โดยที่ชั้นข้อมูลเข้ารับข้อมูลจากอินพุตเวคเตอร์ X และ Y และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

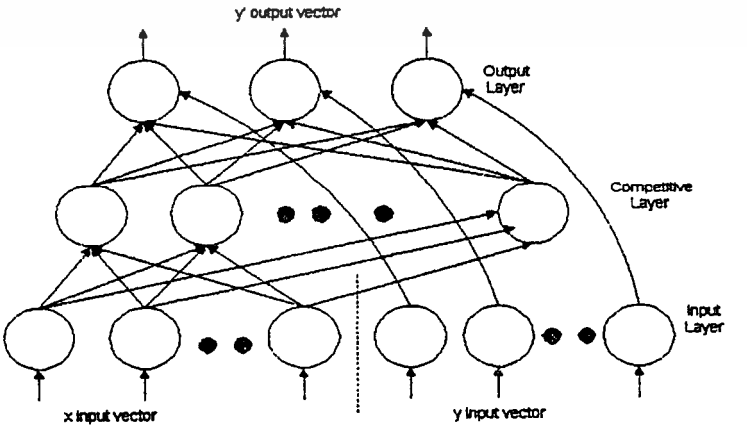
เส้นอินเตอร์คอนเน็กชันที่เป็นผลลัพธ์ของชั้นคอมเพททิทิพจะเชื่อมกับชั้นผลลัพธ์ (output layer) แบบทั่วถึงกันหมดเช่นกัน ชั้นผลลัพธ์นอกจากรับข้อมูลจากชั้นคอมเพททิทิพแล้วยังรับข้อมูลจากอินพุตเวคเตอร์ $\mathbf{X}$ และ $\mathbf{Y}$ ซึ่งการเชื่อมต่อกับอินพุตเวคเตอร์นี้จะแบ่งโหนดของชั้นผลลัพธ์ออกเป็น 2 กลุ่มคือ โหนดที่ให้เวคเตอร์ผลลัพธ์เป็น $\mathbf{X}'$ และ $\mathbf{Y}'$ โดยอินพุตเวคเตอร์ $\mathbf{X}$ เชื่อมกับโหนดที่ให้ผลลัพธ์ $\mathbf{X}'$ และอินพุตเวคเตอร์ $\mathbf{Y}$ เชื่อมกับโหนดที่ให้ผลลัพธ์ $\mathbf{Y}'$ รูปแบบการเชื่อมต่อของชั้นผลลัพธ์กับอินพุตเวคเตอร์เป็นแบบหนึ่งต่อหนึ่ง

จากที่กล่าวข้างต้นการสอนของเคาน์เตอร์พรอพาเกชันมีสองขั้นตอน ซึ่งขณะสอนโครงข่ายชนิดนี้จะมีอินพุตเวคเตอร์ $\mathbf{X}$ และ $\mathbf{Y}$ ปรากฏทั้งสองขั้นตอน ในขั้นตอนที่สองของการสอนโครงข่ายชนิดนี้ค่าถ่วงน้ำหนักของโหนด $\mathbf{X}'$ และ $\mathbf{Y}'$ ในชั้นผลลัพธ์ที่เชื่อมกับชั้นคอมเพททิทิพจะถูกปรับให้มีค่าเท่ากับอินพุตเวคเตอร์ $\mathbf{X}$ และ $\mathbf{Y}$ ที่ต่อกับโหนดในชั้นนี้ตามลำดับ ซึ่งการเรียนในขั้นที่สอนนี้จะขอกล่าวในภายหลัง

หลังจากการสอนโครงข่ายเคาน์เตอร์พรอพาเกชันแบบเต็มรูปแบบเสร็จแล้ว เมื่อนำโครงข่ายไปใช้งานสามารถสร้างผลลัพธ์ที่ได้สองลักษณะกล่าวคือ เมื่ออินพุตเวคเตอร์ $(\mathbf{X},\mathbf{0})$ เข้าสู่โครงข่ายค่าของผลลัพธ์ที่ได้จะเป็น $y'=\Phi(x)$ และเมื่ออินพุตเวคเตอร์ $(\mathbf{0},\mathbf{Y})$ เข้าสู่โครงข่ายค่าของผลลัพธ์ที่ได้จะเป็นค่าของฟังก์ชันแบบอินเวอร์ของ $x'=\Phi^{-1}(y)$

โครงสร้างเคาน์เตอร์พรอพาเกชันแบบแพร่ไปข้างหน้านี้ใช้ประมาณค่าฟังก์ชันแบบต่อเนื่องของ $y=\Phi(x)$ โดยไม่สนใจอินเวอร์ฟังก์ชัน โครงสร้างนี้ประกอบด้วย 3 ชั้น คือ ชั้นข้อมูลเข้า (input layer),ชั้นของคอมเพททิทิพ (Competitive layer) และชั้นของผลลัพธ์ (output layer) ภาพที่ 3.2 แสดงโครงสร้างเคาน์เตอร์พรอพาเกชันแบบแพร่ไปข้างหน้า

ภาพที่ 15



แสดงสถาปัตยกรรมของเคาน์เตอร์พรอพาเกชันแบบแพร่ไปข้างหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

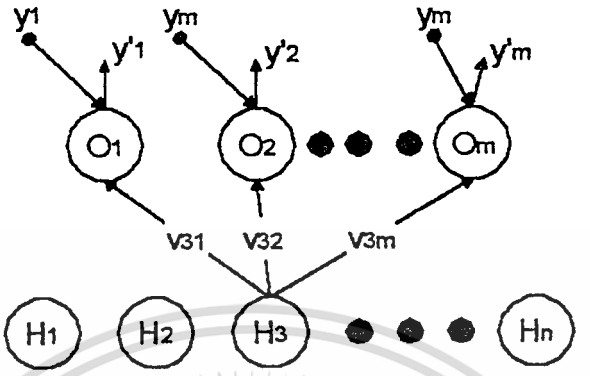
จากภาพที่ 15 จะเห็นว่าโครงสร้างของเคาน์เตอร์พรอพาเกชันแบบแพร่ไปข้างหน้าแตกต่างจากโครงสร้างแบบเต็มรูปแบบตรงที่ลักษณะการเชื่อมต่อในชั้นคอมเพททิทิพและชั้นผลลัพธ์ ซึ่งจากภาพจะเห็นว่าในชั้นคอมเพททิทิพของโครงข่ายแบบแพร่ไปข้างหน้าไม่มีการนำอินพุตเวคเตอร์ Y มาเชื่อมต่อ และในชั้นผลลัพธ์อินพุตเวคเตอร์ที่มาเชื่อมต่อกับชั้นนี้มีอินพุตเวคเตอร์ Y เพียงอย่างเดียว โดยขณะสอนโครงข่ายชนิดนี้ขั้นตอนแรกมี X เป็นอินพุตเวคเตอร์ที่นำมาสู่โครงข่ายเพียงอย่างเดียว และขั้นที่สองมีอินพุตเวคเตอร์ X และ Y ปรากฏทั้งคู่โดยที่อินพุตเวคเตอร์ X ปรากฏที่ชั้นที่รับข้อมูลเข้าและอินพุตเวคเตอร์ Y ปรากฏที่ชั้นผลลัพธ์ ซึ่งการเรียนในขั้นตอนนี้ค่าถ่วงน้ำหนักของเส้นอินเตอร์คอนเน็กชันที่เชื่อมระหว่างชั้นคอมเพททิทิพและชั้นผลลัพธ์จะถูกปรับให้มีค่าเท่ากับอินพุตเวคเตอร์ Y เมื่อมีอินพุตเวคเตอร์ X เข้าสู่โครงข่าย และเมื่อมีการนำโครงข่ายไปประยุกต์ใช้งานข้อมูลที่เข้าสู่โครงข่ายมีรูปแบบเดียวคือ (X,0) และผลลัพธ์ที่ได้จากโครงข่ายอยู่ในรูป $y=\Phi(x)$

โครงสร้างของชั้นคอมเพททิทิพหรือชั้นของโคโฮเนนมีลักษณะการทำงานแบบแข่งขัน กล่าวคือการทำงานของชั้นนี้จะการคัดเลือกโหนดหรือเซลประสาทเทียมภายในชั้นนี้ให้เป็นโหนดที่ active เพียงโหนดเดียวโดยหาผลลัพธ์ของโหนดที่ active นี้มีค่าเท่ากับหนึ่งและให้ผลลัพธ์ของโหนดอื่นมีค่าเท่ากับศูนย์ ซึ่งจะขอกล่าวรายละเอียดในหัวข้อถัดไป จากภาพที่ 16 สมมุติให้ H เป็นชั้นของคอมเพททิทิพซึ่งมีโหนดที่ 3 (H_3) เป็นโหนดที่ active ดังนั้น Z ซึ่งเป็นเวคเตอร์ผลลัพธ์ของชั้นนี้คือ $[0_1,0_2,1_3,0_4,.....,0_n]$

เนื่องจากชั้นคอมเพททิทิพมีหน้าที่แบ่งหรือแยกข้อมูลออกเป็นกลุ่มๆซึ่งในกลุ่มข้อมูลเดียวกันอาจประกอบด้วยโหนดในชั้นคอมเพททิทิพหลายโหนดได้ ด้วยเหตุนี้การสอนในชั้นผลลัพธ์ (Output Layer) ของ CPN เพื่อบอกชื่อกลุ่มหรือระบุชื่อกลุ่มให้กับโหนดในชั้นคอมเพททิทิพ การสอนในชั้นผลลัพธ์ของ CPN ใช้วิธีการเรียนแบบ Groberg learning ซึ่งจะมีการกำหนดอินพุตเวคเตอร์ซึ่งเป็นผลลัพธ์ที่เราต้องการให้โครงข่ายสร้างให้ โดยอินพุตเวคเตอร์นี้จะถูกนำเข้าสู่โครงข่ายในชั้นผลลัพธ์ (output layer) เพื่อให้โครงข่ายปรับค่าถ่วงน้ำหนักของเส้นอินเตอร์คอนเน็กชันที่เชื่อมระหว่างชั้นผลลัพธ์กับชั้นคอมเพททิทิพให้สามารถสร้างผลลัพธ์ที่ตรงกับอินพุตเวคเตอร์ที่นำเข้ามาสอนที่ชั้นผลลัพธ์ การสอนด้วยวิธีนี้ค่าถ่วงน้ำหนักของชั้นผลลัพธ์จะปรับเฉพาะเส้นอินเตอร์คอนเน็กชันเชื่อมต่อกับโหนดที่ active ในคอมเพททิทิพเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 16



แสดงตัวอย่างการเรียนของชั้นผลลัพธ์ของ CPN ด้วยวิธี Groberg learning

ภาพที่ 16 แสดงตัวอย่างการเรียนของชั้นผลลัพธ์ของ CPN โดยมี Y, Y' เป็นอินพุตเวคเตอร์และเอาท์พุตเวคเตอร์ของชั้นนี้ตามลำดับ ให้ H_3 คือโหนดที่ active ของชั้นคอมเพททิทิพ ซึ่งมี V คือเวคเตอร์ของค่าถ่วงน้ำหนักของโหนด H_3 ที่เชื่อมต่อกับชั้นผลลัพธ์ที่ได้รับการเรียน (ปรับค่าถ่วงน้ำหนัก) ซึ่งเวคเตอร์ของค่าถ่วงน้ำหนัก V จะได้รับการเรียนจนกระทั่งมีค่าเท่ากับ Y ดังนั้นเมื่อมีข้อมูลเข้าสู่โครงข่ายแล้วกระตุ้นให้ H_3 active ผลลัพธ์ที่ได้จากโครงข่ายนี้จะเหมือนกับเวคเตอร์ Y ที่นำมาสอนในโครงข่าย

ด้วยลักษณะการเรียนแบบ Groberg learning ในขั้นตอนการสอนชั้นผลลัพธ์เป็นผลให้การทำงานในชั้นนี้ไม่มีการแข่งขันกันระหว่างโหนดต่างๆภายในชั้นของโครงข่าย เนื่องจากค่าถ่วงน้ำหนักของอินเตอร์คอนเน็กชันที่เชื่อมระหว่างชั้นผลลัพธ์กับโหนดที่ active ในคอมเพททิทิพนี้ได้ถูกปรับให้เท่ากับอินพุตเวคเตอร์ที่นำเข้ามาสอน ดังนั้นเมื่อนำข้อมูลมาวิเคราะห์ในโครงข่ายค่าเอาท์พุตเวคเตอร์ที่ได้จึงให้ผลลัพธ์เหมือนค่าอินพุตเวคเตอร์ที่นำมาสอนในชั้นผลลัพธ์[7,8]

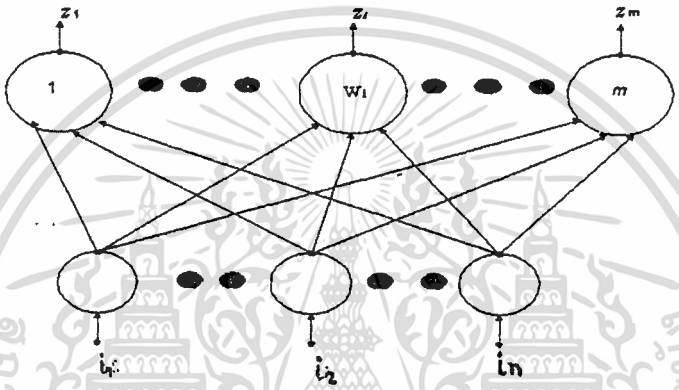
หลักการของ CPN ที่ใช้ในงานวิจัย

โครงสร้างของ CPN ที่นำมาประยุกต์ใช้ในงานวิจัยนี้คือ โครงสร้างของเคาน์เตอร์พรอพาเกชันแบบแพร่ไปข้างหน้า เนื่องจากงานวิจัยนี้นำ CPN ใช้ในแยกแยะตัวอักษรภาษาไทย ซึ่งแต่ละโหนดจะต้องจดจำลักษณะตัวแบบอักษรเพียงประเภทเดียวเท่านั้น ดังนั้นเราจึงสามารถที่จะระบุได้ว่าโหนดใดในชั้นของคอมเพททิทิพจดจำตัวแบบอักษรประเภทอะไรได้ จากเหตุผลข้างต้นนี้โครง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สร้างของ CPN ที่ใช้ในงานจึงประกอบด้วย 2 ชั้นคือ ชั้นข้อมูลเข้า (Input Layer) และชั้นคอมเพททิทิพ (Competitive Layer) ซึ่งจะช่วยลดเวลาในการคำนวณและหน่วยความจำที่ใช้ในการเก็บค่าถ่วงน้ำหนัก เนื่องจากโครงสร้างของ CPN ที่ใช้ในงานวิจัยนี้ประกอบด้วย 2 ชั้น ดังนั้นงานวิจัยนี้จึงขอกล่าวถึงโครงสร้างแต่เพียงที่ใช้ในงานวิจัยเท่านั้น

ภาพที่ 17



แสดงโครงสร้างของ CPN ที่ใช้ในงานวิจัย

ภาพที่ 17 แสดงโครงสร้างของ CPN ที่ใช้ในงานวิจัยซึ่งประกอบด้วยชั้นข้อมูลเข้า (Input Layer) จำนวน n โหนด แต่ละโหนดรับข้อมูลจากอินพุตเวคเตอร์ $\mathbf{I} = (i_1, i_2, \ldots, i_n)$ แล้วทำการกระจาย component ของอินพุตเวคเตอร์ $\mathbf{I}$ ให้กับชั้นคอมเพททิทิพ (Competitive Layer) จำนวน m โหนด สมการที่ 3.1 แสดงการหาค่า net input ของโหนดที่ i ในชั้นคอมเพททิทิพของโครงข่ายที่ใช้ในงานวิจัย

$$\mathrm{net}_i = \sum_j^m i_j w_{ij} \qquad \text{.....(3.1)}$$

โดยที่ i_j : ค่าอิลิเมนต์ (element) ลำดับที่ j ของอินพุตเวคเตอร์ $\mathbf{I}$

w_{ij} : ค่าอิลิเมนต์ลำดับที่ j ของเวคเตอร์ของค่าถ่วงน้ำหนักของโหนดที่ i

และค่าของเอาท์พุตเวคเตอร์ของชั้นคอมเพททิทิพ คือ $\mathbf{Z} = (z_1, z_2, \ldots, z_m)$ จากที่กล่าวข้างต้นชั้นคอมเพททิทิพมีลักษณะแบบแข่งขันซึ่งจะคัดเลือกโหนดที่ active เพียงโหนดเดียวเท่านั้น โดยการคัดเลือกจะนำโหนดที่มีค่า net input มากที่สุดเป็นโหนดที่ active และให้ผลลัพธ์ที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโหนดนี้มีค่าเท่ากับ 1 และให้ค่าผลลัพธ์ของโหนดอื่นๆมีค่าเท่ากับ 0 หมด ดังนั้นเมื่อโหนด i มีค่า net input มากที่สุดค่าของ z_i จึงมีค่าเท่ากับ 1 และเรียกโหนดนี้ว่า โหนดผู้ชนะ (winner Node) ส่วนค่าของโหนดอื่นๆจะถูกกำหนดให้มีค่าเป็น 0 เพราะฉะนั้นค่าเอาท์พุตเวคเตอร์ $\mathbf{Z}$ นี้ คือ $[0_1,0_2,0_3,...1_k,.....0_m]$ ซึ่งการหาค่า $\mathbf{Z}$ แสดงด้วยสมการที่ 3.2

$$z_i = \begin{cases} 1 & net_i > net_j \text{ for all } j \neq i \\ 0 & otherwise \end{cases} \qquad \text{.....(3.2)}$$

การที่โหนดมีค่า net input มีมากที่สุดแสดงให้เห็นว่าค่าถ่วงน้ำหนักของอินเตอร์คอนเน็กชันที่เชื่อมกับโหนดนั้น (weight vector) มีระยะที่ใกล้เคียงกับอินพุตเวคเตอร์ที่เข้าสู่โครงข่ายที่มากสุด ซึ่งในชั้นของคอมเพตทิทิพนี้แต่ละโหนดจะแสดงถึงกลุ่มข้อมูลแบบต่างๆที่ถูกแบ่งออกมาในขณะสอน โดยที่จำนวนของโหนดของชั้นนี้จะขึ้นอยู่กับจำนวนของกลุ่มข้อมูลที่มีอยู่

ข้อมูลเข้าสู่โครงข่าย CPN ที่ใช้ในงานวิจัยอยู่ในรูปของภาพตัวอักษรซึ่งประกอบด้วยค่า 1 และ 0 ซึ่งค่าทั้งสองนี้แทนค่าเนื้อตัวอักษรและพื้นตัวอักษรตามลำดับ รูปที่ 25 แสดงตัวอย่างการนำอินพุตภาพตัวอักษรเข้าสู่โครงข่ายที่ใช้ในงานวิจัย และลักษณะการวางภาพตัวอักษรบนอินพุตเมตริกจะวางที่ตำแหน่งชิดมุมบนซ้ายของอินพุตเมตริกแสดงดังภาพที่ 40

วิธีการสอน CPN

การสอนของ CPN ในชั้นคอมเพททิทิพ[2] เวคเตอร์ของค่าถ่วงน้ำหนัก (weight vector) ของแต่ละโหนดในชั้นคอมเพททิทิพจะทำหน้าที่เสมือนเป็นตัวแทนหรือตัวอย่างของกลุ่มข้อมูลที่ถูกเก็บไว้ในโครงข่ายโครงข่าย โดยที่กลุ่มข้อมูลนี้โครงข่ายเป็นผู้แบ่งซึ่งการสอนในชั้นคอมเพททิทิพของ CPN จะคัดเลือกโหนดภายในชั้นนี้ที่มีระยะห่างระหว่างเวคเตอร์ของค่าถ่วงน้ำหนักกับอินพุตเวคเตอร์ (input vector) ที่มีระยะใกล้กันมากที่สุดเพียงโหนดเดียวที่นำมาสอน ซึ่งโหนดนี้คือ โหนดผู้ชนะของชั้นคอมเพททิทิพนั้นเอง

เนื่องจากการสอน CPN มีลักษณะเป็นแบบการฝึกแบบไม่มีผู้สอน ดังนั้นในขณะสอนโครงข่ายจึงไม่มีการกำหนดผลลัพธ์เป้าหมาย (target output) โดยลักษณะการปรับค่าถ่วงน้ำหนักของโหนดในชั้นคอมเพททิทิพจะปรับให้มีลักษณะเหมือนกับตัวอย่างข้อมูลที่มาตกในโหนดนั้น (โหนดผู้ชนะ) สมการที่ 3.3 แสดงสมการที่ใช้สอนโครงข่าย

เอกสารนี้มีลิขสิทธิ์ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$W(t+1) = W(t) + \alpha (I - W(t)) \quad \text{.....(3.3)}$$

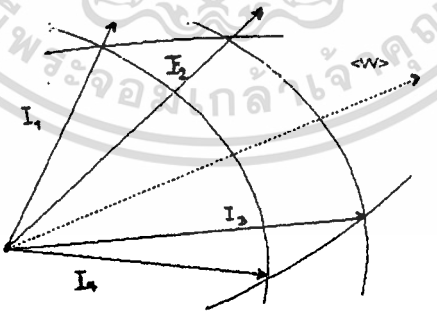
โดยที่ I : ค่าอินพุตเวคเตอร์ (input vector) ที่เข้าสู่โครงข่าย

α : ค่า learning rate ($0 < \alpha < 1$)

W : ค่าเวคเตอร์ของค่าถ่วงน้ำหนักที่เวลา t ของโหนดผู้ชนะของชั้นคอมเพททิทิพ

การสอนโครงข่ายจะทำจนกระทั่งโครงข่ายสามารถปรับตัวเองจนสามารถแยกหรือแบ่งตัวอย่างที่นำมาสอนที่มีลักษณะเหมือนกันออกเป็นกลุ่มๆได้ ซึ่งเวคเตอร์ของค่าถ่วงน้ำหนักของโหนดที่ได้รับการสอนมีลักษณะเป็นค่าเฉลี่ยของตัวอย่างข้อมูลที่นำมาสอนแล้วนั้นมาตกในโหนดนั้น ภาพที่ 18 แสดงทิศทางเวคเตอร์ของค่าถ่วงน้ำหนักเฉลี่ยที่ได้จากการสอนของกลุ่มอินพุตเวคเตอร์ I

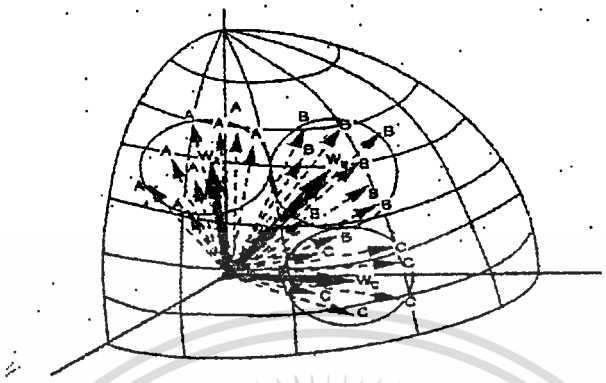
ภาพที่ 18



แสดงลักษณะของทิศทางของค่าถ่วงน้ำหนักเฉลี่ยที่ได้จากการสอนโครงข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 19

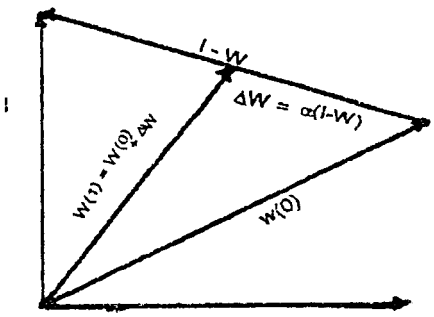


แสดงตัวอย่างของทิศทางของค่าถ่วงน้ำหนักเฉลี่ย
ที่ใช้ในการแบ่งกลุ่มตัวอย่างข้อมูล A,B,C

ภาพที่ 19 แสดงตัวอย่างทิศทางของค่าถ่วงน้ำหนักเฉลี่ยหลังจากการสอนเพื่อใช้แยกกลุ่มข้อมูล A,B และ C โดยที่วงกลมแสดงขอบเขตของข้อมูลแต่ละกลุ่ม และลูกสอนใหญ่สีเข้มแทนทิศทางของค่าถ่วงน้ำหนักเฉลี่ยที่ได้จากการสอนของแต่ละกลุ่ม

จากสมการที่ 3.3 ทิศทางการปรับของเวคเตอร์ของค่าถ่วงน้ำหนักในชั้นคอมเพททิทิพจะเปลี่ยนตามทิศของตัวอย่างข้อมูลที่เข้ามาสอนโครงข่าย ซึ่งระยะการปรับของเวคเตอร์ของค่าถ่วงน้ำหนักเมื่อมีการนำตัวอย่างข้อมูลเข้าสู่โครงข่ายขึ้นอยู่กับอยู่ค่า α โดยที่ถ้าค่า α มากทิศทางการปรับของ W ก็จะลู่เข้าหา I มาก ภาพที่ 20 แสดงทิศทางการปรับทิศทางของเวคเตอร์ของค่าถ่วงน้ำหนัก W เมื่อมีตัวอย่างข้อมูล I เข้าสู่โครงข่าย

ภาพที่ 20



แสดงทิศทางการปรับเวคเตอร์ของค่าถ่วงน้ำหนัก
เมื่อมีข้อมูล I เข้ามาในโครงข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากระยะการปรับของเวคเตอร์ของค่าถ่วงน้ำหนัก $\mathbf{W}$ ขึ้นอยู่กับค่า α และทิศทางของข้อมูล $\mathbf{I}$ ดังนั้นถ้า α มีค่ามากทิศทางการปรับของเวคเตอร์ของค่าถ่วงน้ำหนัก $\mathbf{W}$ ก็จะปรับมาทางข้อมูล $\mathbf{I}$ มากจึงเป็นผลให้โครงข่ายสามารถจดจำตัวอย่าง (อินพุตเวคเตอร์ที่นำเข้าสอน) ที่กำลังสอนได้อย่างรวดเร็ว และเช่นเดียวกันโครงข่ายก็จะลืมตัวอย่างที่เคยสอนอย่างรวดเร็ว ในทางกลับกันถ้า α มีค่าน้อยก็จะจดจำตัวแบบที่นำเข้ามาสอนโครงข่ายได้ช้า

อัลกอลิทึมที่ใช้ในการสอนชั้นคอมเพททิทิพของ CPN อัลกอลิทึม (algorithm) ที่ใช้ในการสอน CPN ที่กล่าวถึงนี้เป็นส่วนของคอมเพททิทิพ[3]

Step 0. initialize weights,learning rates (α)

Step 1. While stopping condition is False,do steps 2-7

Step 2. for each training input I, do step 3-5

Step 3. Set I input activation to vector $\mathbf{I}$.

Step 4. Find winning cluster unit

$$net_i = \sum_j^m i_j w_{ij}$$

call index j

Step 5. update weights for unit z_j

$$\mathbf{W}_j(t+1) = \mathbf{W}_j(t) + \alpha(\mathbf{I} - \mathbf{W}_j(t))$$

Step 6. Reduce learning rate (α)

Step 7. if weight is average or learning rate <= 0

stopping condition is True

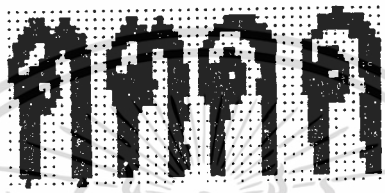
ปัญหาการใช้ CPN แยกแยะตัวพิมพ์อักษรภาษาไทย

ปัญหาการเรียนผิดโหนด เนื่องจากการสอนของโครงข่าย CPN เลือกปรับค่าถ่วงน้ำหนักของโหนดที่มีค่า net input มากที่สุดเพียงโหนดเดียว ดังนั้นอาจเกิดปัญหาการเรียนผิดโหนด กรณีแรกเนื่องจากตัวอักษรภาษาไทยมีรูปร่างคล้ายกัน เช่น ก,ถ,ภ,ศ,ด,ค,ศ,ด และภาพของตัวอักษรเมื่อผ่านการสแกนแล้วจะยิ่งทำให้ลักษณะภาพมีความคล้ายกันมาก ภาพที่ 21 แสดงตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของภาพคล้ายกันของ ค,ค ที่ได้จากการสแกนภาพ ซึ่งจากตัวอย่างภาพแสดงให้เห็นว่าระยะห่างระหว่างของขอบเขตของกลุ่มตัวอักษรมีระยะใกล้กันมาก จากภาพที่ 19 เป็นที่แสดงให้เห็นถึงตัวอย่างกรณีขอบเขตของข้อมูลกลุ่ม A,B และ C มีระยะใกล้กัน ซึ่งจากภาพจะเห็นว่ากลุ่ม B และ C มีระยะที่ใกล้เคียงกันมากโดยสังเกตจากการกระจายของกลุ่มข้อมูล

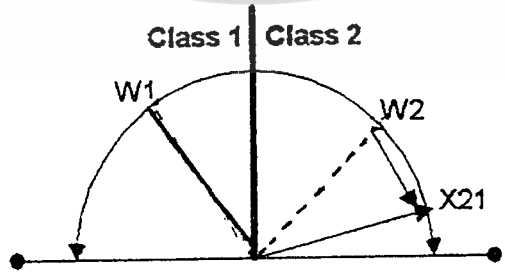
ภาพที่ 21



แสดงความคล้ายของภาพที่ได้จากการสแกนของตัวอักษร ค,ค

ภาพที่ 22 แสดงตัวอย่างปัญหาการปรับทิศทางของเวคเตอร์ของค่าถ่วงน้ำหนัก กรณีระยะห่างระหว่างขอบเขตกลุ่มที่ 1 และ 2 ใกล้กัน โดยสมมุติให้ **W1** และ **W2** ถูกสอนเพื่อใช้ในการแยกกลุ่มที่ 1 และ 2 เริ่มแรกให้มีตัวอย่างข้อมูล **X21** ซึ่งอยู่ในกลุ่มที่ 2 เข้าสู่โครงข่าย ซึ่งค่า **W2** จะถูกปรับแสดงทิศทางในภาพที่ 22.1 จากนั้นมี **X22** ของกลุ่มที่ 2 เข้าสู่โครงข่ายอีกจะเห็นว่า **W1** ซึ่งอยู่ในกลุ่มที่ 1 จะถูกเรียนดังแสดงในภาพที่ 22.2 เนื่องจากทิศทางของ **W1** เมื่อเทียบกับ **W2** เมื่อมี **X22** เข้าสู่โครงข่ายแล้วมีระยะใกล้ **W2** มากกว่าจึงทำให้มีการเรียนผิดโหนด

ภาพที่ 22



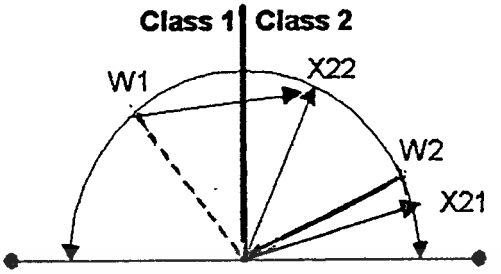
22.1) แสดงทิศทางการปรับเวคเตอร์ของค่าถ่วงน้ำหนักของ W2 เมื่อมีข้อมูล X21

แสดงปัญหาการเรียนผิดโหนดของ CPN

กรณีระยะห่างระหว่างขอบเขตของกลุ่มข้อมูลใกล้กัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 22 (ต่อ)



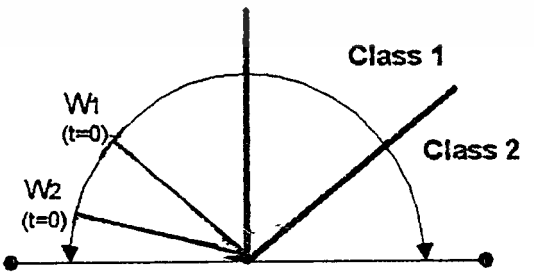
22.2) แสดงทิศทางการปรับเวคเตอร์ของค่าถ่วงน้ำหนักของ W1 เมื่อมีข้อมูล X22

แสดงปัญหาการเรียนผิดโหนดของ CPN

กรณีระยะห่างระหว่างขอบเขตของกลุ่มข้อมูลใกล้กัน

กรณีที่สองเกิดจากการกำหนดทิศทางของค่าถ่วงน้ำหนักเริ่มต้นไม่ดี เช่น เราต้องการสอนให้โครงข่ายสามารถแยกข้อมูล 2 กลุ่ม คือ X1 และ X2 ซึ่งกำหนดค่าเริ่มต้นของค่าถ่วงน้ำหนักคือ W1 และ W2 แสดงดังภาพที่ 23.1 จากนั้นนำ X1 มาสอนให้กับโครงข่ายซึ่งเวคเตอร์ของค่าถ่วงน้ำหนัก W1 จะถูกปรับเนื่องจากมีระยะห่างใกล้กับ X1 ที่สุด แสดงดังภาพที่ 23.2 และเช่นเดียวกันเมื่อนำ X2 ซึ่งเป็นข้อมูลของกลุ่มที่ 2 เข้าสู่โครงข่ายเวคเตอร์ของค่าถ่วงน้ำหนัก W1 ก็จะถูกปรับอีกเนื่องจากมีระยะใกล้กับ X2 ที่สุดแสดงดังภาพที่ 23.3 ซึ่งเมื่อทำการสอนไปเลื่อยๆค่าของเวคเตอร์ของค่าถ่วงน้ำหนัก W2 ก็จะถูกเรียนเพียงโหนดเดียว[7]

ภาพที่ 23



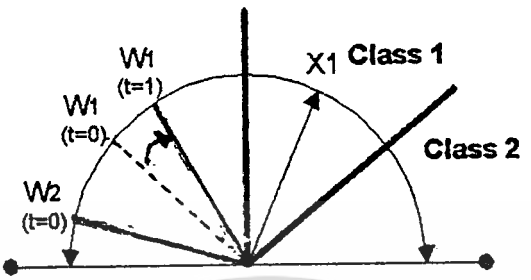
23.1) แสดงการกำหนดทิศทางเริ่มต้นของค่าถ่วงน้ำหนักของ W1 และ W2

แสดงปัญหาการเรียนผิดโหนดของ CPN

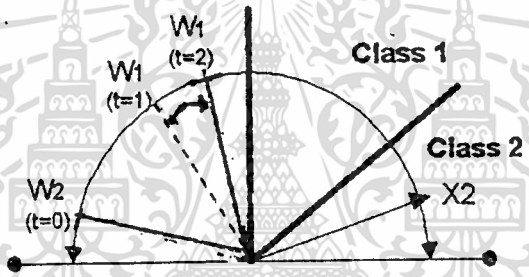
กรณีกำหนดทิศทางเริ่มต้นของค่าถ่วงน้ำหนักไม่ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 23 (ต่อ)



23.2) แสดงการปรับทิศทางของค่าถ่วงน้ำหนัก W1
เมื่อมีข้อมูล X1 ของกลุ่มที่ 1 เข้าสู่โครงข่าย



23.3) แสดงการปรับทิศทางของค่าถ่วงน้ำหนัก W1
เมื่อมีข้อมูล X2 ของกลุ่มที่ 2 เข้าสู่โครงข่าย

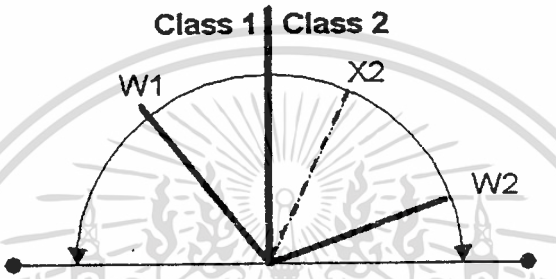
แสดงปัญหาการเรียนผิดโหนดของ CPN
กรณีกำหนดทิศทางเริ่มต้นของค่าถ่วงน้ำหนักไม่ดี

ปัญหาการหาค่าถ่วงน้ำหนักเฉลี่ย[2] จากภาพที่ 21 แสดงตัวอย่างภาพตัวอักษร ค,ค ที่ได้จากการสแกนพบว่ามีหลายรูปแบบมาก ซึ่งแสดงให้เห็นว่าขอบเขตของข้อมูลของแต่ละกลุ่มตัวอักษรมีกว้างมาก ดังนั้นการสอน CPN เพื่อให้ได้ค่าทิศทางของค่าถ่วงน้ำหนักเฉลี่ยจึงเป็นสิ่งที่ยากมาก เพราะขึ้นอยู่กับจำนวนครั้งในการสอนโครงข่ายหรือจำนวนตัวอย่างที่นำมาใช้ในการสอน ภาพที่ 24 แสดงตัวอย่างปัญหาการหาค่าทิศทางของค่าถ่วงน้ำหนักเฉลี่ยผิดของ W2 ซึ่ง W1,W2 เป็นทิศทางของค่าถ่วงน้ำหนักที่ได้จากการสอนจนสามารถแยกกลุ่มข้อมูลที่ 1 และ 2 ตามลำดับ แต่เมื่อมี X2 ซึ่งเป็นข้อมูลที่ไม่ได้ถูกนำมาสอนเข้าสู่โครงข่ายจะถูกจำแนกให้เป็นของกลุ่มที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากระยะห่างระหว่าง X2 กับ W1 น้อยกว่าระยะห่างระหว่าง X2 กับ W2 ซึ่งผิดเนื่องจาก X2 เป็นข้อมูลของกลุ่มที่ 2

ภาพที่ 24



แสดงปัญหาการหาค่าทิศทางของค่าถ่วงน้ำหนักผิด

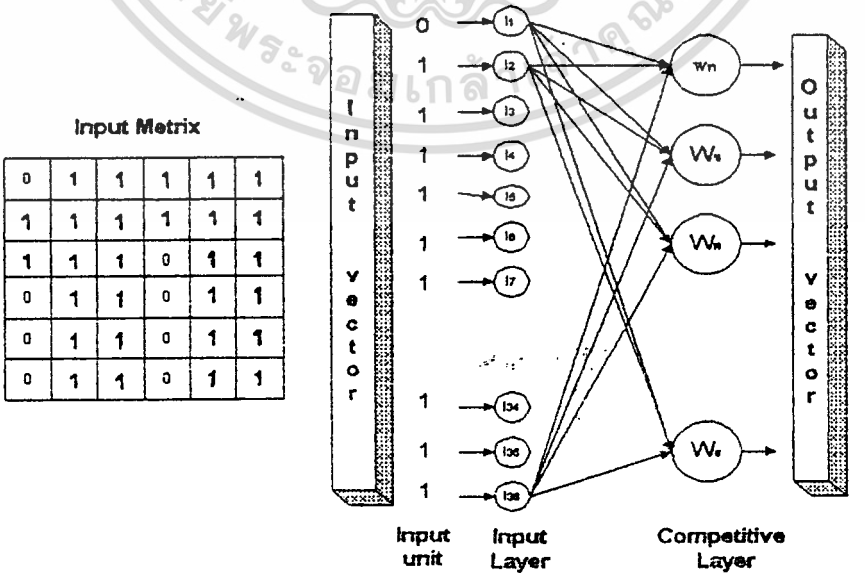
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การประยุกต์ CPN เพื่อการแยกแยะตัวพิมพ์อักษรภาษาไทย

งานวิจัยนี้นำ CPN มาประยุกต์ใช้ในงานแยกแยะตัวพิมพ์อักษรภาษาไทย จากที่กล่าวในบทที่ 3 โครงสร้างของ CPN ที่ใช้ในงานวิจัยประกอบด้วย ชั้นรับข้อมูลเข้า (Input Layer) และชั้นคอมเพททิทิพ (Competitive Layer) ในการนำ CPN มาประยุกต์ใช้นี้ข้อมูลที่นำเข้าสู่โครงข่ายเป็นข้อมูลภาพของตัวอักษรที่ที่ประกอบด้วยจุดภาพแสดงดังอินพุตเมตริก (input metrix) ภาพที่ 25 โดย 1 แทนเนื้อตัวอักษร และ 0 แทนพื้นตัวอักษร และลักษณะการวางภาพตัวอักษรบนอินพุตเมตริกจะวางที่ตำแหน่งชิดมุมบนซ้ายของอินพุตเมตริกแสดงดังภาพที่ 40 ภาพที่ 25 แสดงตัวอย่างการนำข้อมูลภาพตัวอักษรที่อยู่ในรูปเมตริก (metrix) ขนาด 6x6 เข้าสู่โครงข่ายใช้ในงานวิจัย

ภาพที่ 25



แสดงการนำข้อมูลภาพตัวอักษรเข้าสู่โครงข่าย CPN ที่ใช้ในงานวิจัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

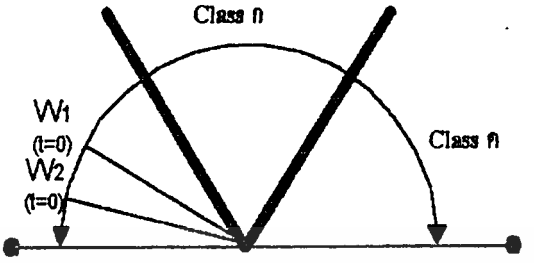
จากที่กล่าวข้างต้นก่อนการนำโครงข่ายมาประยุกต์ใช้งานจะต้องทำการฝึกสอนโครงข่ายก่อนเพื่อให้โครงข่ายมีความรู้เกี่ยวกับปัญหานั้นๆ ซึ่งในบทที่ 3 แสดงให้เห็นถึงปัญหาในการฝึกสอนของ CPN บทนี้จะกล่าวถึงวิธีการแก้ปัญหาและขั้นตอนการฝึกสอน CPN เพื่อนำมาประยุกต์ใช้งานแยกแยะตัวพิมพ์อักษรภาษาไทย ซึ่งขั้นตอนต่างๆแสดงดังรายละเอียดที่จะกล่าวต่อไปนี้

<u>การฝึกสอน CPN เพื่อการแยกแยะตัวพิมพ์อักษรภาษาไทย</u>

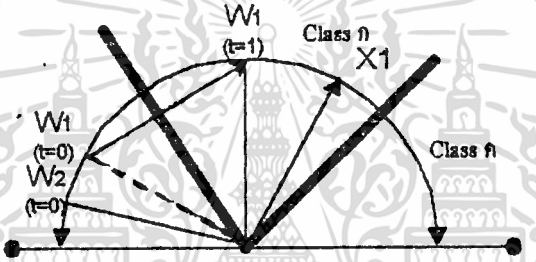
1.การกำหนดค่าถ่วงน้ำหนักเริ่มต้น ก่อนการสอนโครงข่ายนั้นจะต้องทำการกำหนดค่าถ่วงน้ำหนักเริ่มต้นก่อน ซึ่งการกำหนดค่าเริ่มต้นด้วยวิธีการสุ่ม (random) นี้ในการสอนโครงข่าย CPN เพื่อใช้ในการแยกแยะตัวอักษรภาษาไทยอาจเกิดปัญหาดังที่กล่าวในบทที่ 3 เรื่องปัญหาการเรียนผิดโหนดกรณีการกำหนดค่าถ่วงน้ำหนักเริ่มต้นไม่ดี เช่น เมื่อต้องการสอนโครงข่ายให้แยกแยะตัวอักษร ก,ค ขั้นตอนแรกกำหนดค่าถ่วงน้ำหนักเริ่มต้นวิธีการสุ่ม ซึ่งทิศทางของค่าถ่วงน้ำหนักเริ่มต้นแสดงดังภาพที่ 26.1 จากนั้นนำตัวอย่างข้อมูล ก มาสอนให้กับโครงข่าย จากภาพที่ 26.1 โหนดที่มีทิศทางของค่าถ่วงน้ำหนักใกล้กับตัวอักษร ก จะปรับตัวเองให้มีลักษณะเหมือน ก ภาพที่ 26.2 แสดงทิศทางของการปรับของค่าถ่วงน้ำหนัก W1 เมื่อมีการนำ ก มาสอนให้กับโครงข่าย หลังจากนั้นนำตัวอย่างข้อมูล ค มาสอน (มีลักษณะเหมือน ก ที่ส่วนขาของตัวอักษร) จากภาพที่ 26.2 จะเห็นว่าโหนดที่ตัวอย่างข้อมูล ก ไปตกมีทิศทางใกล้กว่าโหนดอื่น ดังนั้นค่าถ่วงน้ำหนักของโหนดที่ตัวอย่างข้อมูล ก ตกจึงได้รับสอนแสดงดังภาพที่ 26.3 ซึ่งจากรูปจะเห็นว่าทิศทางค่าถ่วงน้ำหนักของโหนดที่ตัวอักษร ก และ ค ตกมีระยะกึ่งกลางระหว่างข้อมูลทั้งสองกลุ่ม ดังนั้นเมื่อมีการสอนในรอบต่อๆไปแล้วก็จะมีตัวอักษรมาตกที่โหนดนี้เพียงโหนดเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

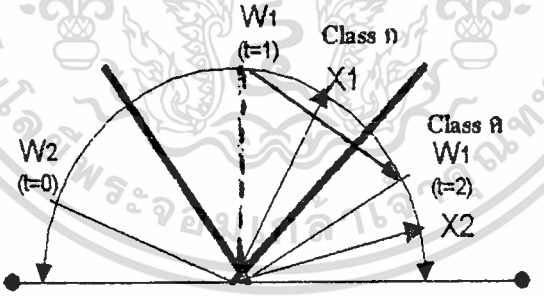
ภาพที่ 26



26.1) แสดงทิศทางการกำหนดค่าถ่วงน้ำหนักเริ่มต้นด้วยวิธีสุ่ม



26.2) แสดงทิศทางการปรับค่าถ่วงน้ำหนัก W1 เมื่อมี X1 เข้าสู่โครงข่าย



26.3) แสดงทิศทางการปรับค่าถ่วงน้ำหนัก W1 เมื่อมี X2 เข้าสู่โครงข่าย

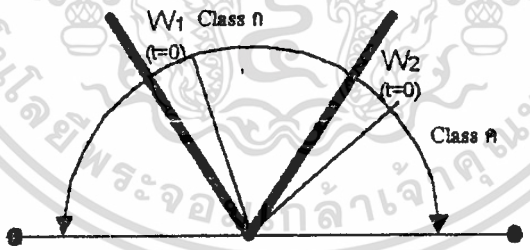
แสดงทิศทางการปรับค่าถ่วงน้ำหนักของการสอนโครงข่ายให้แยกแยะ ก และ ค โดยใช้การกำหนดเริ่มต้นค่าถ่วงน้ำหนักแบบสุ่ม (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1. กำหนดค่าถ่วงน้ำหนักเริ่มต้นโดยใช้ตัวอย่างข้อมูลที่ผ่านการนอล์มอร์-ไลซ์มาเป็นต้นแบบ จากปัญหาการกำหนดค่าถ่วงน้ำหนักเริ่มต้นด้วยวิธีการสุ่มข้างต้นจะสังเกตเห็นว่าทิศทางของค่าถ่วงน้ำหนักที่กำหนดไม่มีการกระจายไปยังข้อมูลกลุ่มต่างๆ งานวิจัยนี้ใช้วิธีการนำตัวอย่างข้อมูลที่นำมาสอนโครงข่ายมาเป็นต้นแบบ (prototype) ในการกำหนดค่าถ่วงน้ำหนักเริ่มต้น ซึ่งจะทำให้ทิศทางของค่าถ่วงน้ำหนักเริ่มต้นกระจายไปยังกลุ่มข้อมูลต่างๆซึ่งช่วยแก้ปัญหาการเรียนผิดโหนดได้

จากปัญหาที่ยกตัวอย่างในกรณีการสอนโครงข่ายให้จดจำตัวอักษร ก และ ค ข้างต้น เมื่อมีการกำหนดค่าถ่วงน้ำหนักเริ่มต้นด้วยวิธีการนำตัวอย่างที่นำมาสอนมาเป็นต้นแบบ แสดงดังภาพที่ 27.1 จากนั้นนำตัวอย่างข้อมูล ก มาสอนให้กับโครงข่าย ค่าถ่วงน้ำหนักของโหนดที่นำ ก เป็นต้นแบบจะถูกปรับเพราะมีทิศทางใกล้ที่สุดแสดงดังภาพที่ 27.2 หลังจากนั้นนำตัวอย่างข้อมูล ค มาสอนซึ่งค่าถ่วงน้ำหนักของโหนดที่นำ ค มาเป็นต้นแบบ จะถูกปรับเช่นเดียวกันเพราะมีทิศทางใกล้ที่สุดแสดงดังภาพที่ 27.3

ภาพที่ 27

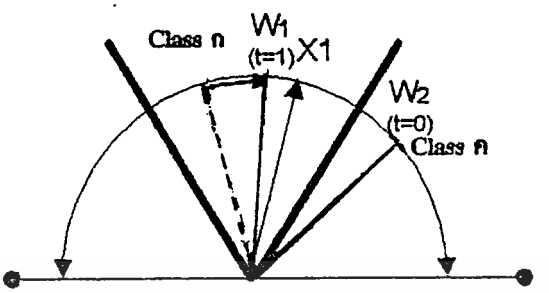


27.1) แสดงทิศทางการกำหนดค่าถ่วงน้ำหนักเริ่มต้น
ด้วยการใช้ตัวอย่างข้อมูลเป็นตัวแบบ

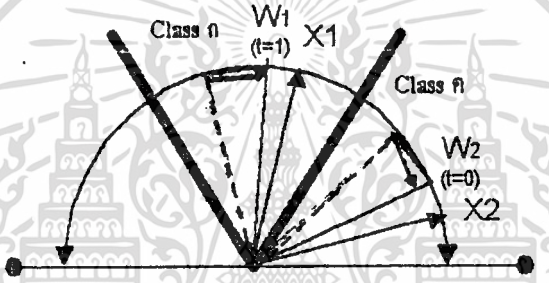
แสดงทิศทางการปรับค่าถ่วงน้ำหนักของการสอนโครงข่าย
ให้แยกแยะ ก และ ค โดยใช้ตัวอย่างข้อมูลเป็นตัวแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 27 (ต่อ)



27.2) แสดงทิศทางการปรับค่าถ่วงน้ำหนัก W1 เมื่อมี X1 เข้าสู่โครงข่าย



27.3) แสดงทิศทางการปรับค่าถ่วงน้ำหนัก W2 เมื่อมี X2 เข้าสู่โครงข่าย

แสดงทิศทางการปรับค่าถ่วงน้ำหนักของการสอนโครงข่าย
ให้แยกแยะ ก และ ค โดยใช้ตัวอย่างข้อมูลเป็นตัวแบบ

แต่เนื่องจากลักษณะตัวอักษรของภาษาไทยมีความคล้ายกัน ดังนั้นการกำหนดให้ค่าถ่วงน้ำหนักเริ่มต้นมีค่าเท่ากับตัวอย่างข้อมูลที่อยู่ในรูป 1 และ 0 อาจทำให้เกิดปัญหาการแยกตัวอักษร เช่น กรณีของ ถ หรือ ฤ ซึ่งจะสังเกตเห็นว่าส่วนหัวของตัวอักษรเหมือนกัน ดังนั้นเมื่อมีข้อมูล ถ เข้าสู่โครงข่ายค่า net input ของโหนด ฤ อาจเป็นผู้ชนะได้ ดังนั้นการกำหนดค่าถ่วงน้ำหนักเริ่มต้นจึงใช้ค่าของอินพุตเวคเตอร์ (input vector) ที่ผ่านการนอร์มอร์ไลซ์ (normalized) ด้วยสมการที่ 4.1 [2]

$$w_i = 1 / \sum_{j=1}^{n} i_j \qquad \text{เมื่อ } i_i = 1$$

$$w_i = 0 \qquad \text{เมื่อ } i_i = 0 \qquad \text{.....(4.1)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ i_j : อิลิเมนต์ลำดับที่ j ของอินพุตเวคเตอร์ I ที่นำมากำหนดค่าถ่วงน้ำหนักเริ่มต้น W และ j มีค่าตั้งแต่ 1 ถึง n

w_i : อิลิเมนต์ลำดับที่ i ของเวคเตอร์ของค่าถ่วงน้ำหนักเริ่มต้น W

และเรียก $(w_1, w_2, \ldots, w_n)^t$ ซึ่งเป็นเวคเตอร์ของค่าถ่วงน้ำหนักเริ่มต้นว่า **reflectance pattern** โดยที่ $\sum_{j=1}^{n} w_j = 1$ การทำนอล์มอร์ไลซ์นี้จะทำให้ค่าอิลิเม็นต์ของอินพุตเวคเตอร์มีความสัมพันธ์กับอินพุตเวคเตอร์ตัวนั้น

1.2. กำหนดให้พื้นตัวอักษรของค่าถ่วงน้ำหนักมีค่าติดลบ เนื่องจากโครงสร้างของ CPN ที่ทำหน้าที่แยกแยะข้อมูลมีเพียงชั้นเดียว (ไม่นับชั้นรับข้อมูล) คือชั้นคอมเพททิทิพ ดังนั้นจึงไม่มีส่วนใดของโครงข่ายที่ช่วยยับยั้งหรือบอกว่าข้อมูลที่เข้ามาตกในโหนดต่างๆนั้นไม่ใช่ข้อมูลของโหนดตนเอง ปัญหาที่เห็นได้ชัดเจนที่สุดก็คือ กรณีการแยกแยะตัวอักษรระหว่าง ฉ,ฤ แม้ว่าจะใช้วิธีการทำนอร์มอล์ไลซ์ในการกำหนดค่าถ่วงน้ำหนักเริ่มต้นแล้ว แต่ก็ยังไม่สามารถแก้ปัญหาได้เพราะว่าปริมาณเนื้อตัวอักษรของ ฉ มีน้อยกว่า ฤ จึงเป็นผลให้ค่าถ่วงน้ำหนักที่เป็นเนื้อตัวอักษรของโหนดที่นำ ฉ มากำหนดค่าเริ่มต้นมีค่ามากกว่า (ต่อไปนี้ข้อใช้คำว่า โหนด ฉ) ภาพที่ 28.1 แสดงลักษณะของค่าถ่วงน้ำหนักเริ่มต้นที่กำหนดโดยวิธีการใช้ต้นแบบที่ผ่านการการนอร์มอล์ไลซ์ในหัวข้อ 1.1 มากำหนดค่าถ่วงน้ำหนักของ ฉ และ ฤ ดังนั้นเมื่อมีข้อมูลคืออินพุตเวคเตอร์ ฤ เข้ามาสอนในโครงข่ายแสดงดังภาพที่ 28.2 ค่าถ่วงน้ำหนักเริ่มต้นของโหนด ฉ อาจเป็นผู้ชนะได้ จากค่าถ่วงน้ำหนักของภาพที่ 28.1 เมื่อนำอินพุตเวคเตอร์ ฤ ในภาพที่ 28.2 มาซ้อนทับ (mapping) จะเห็นว่าค่า net input ของโหนดที่กำหนดค่าถ่วงน้ำหนักเริ่มต้นด้วย ฉ มีค่าเท่ากับ 1 แต่ค่า net input ของโหนดที่กำหนดค่าถ่วงน้ำหนักเริ่มต้นด้วย ฤ มีค่าน้อยกว่า 1 (net input ของโหนด ฤ เท่ากับ 0.947) กรณีนี้ถ้ากำหนดให้พื้นของตัวอักษรที่นำมาเป็นต้นแบบในการกำหนดค่าถ่วงน้ำหนักเริ่มต้นมีค่าติดลบจะช่วยยับยั้งส่วนที่เกินของอินพุตเวคเตอร์ ฤ ในค่าถ่วงน้ำหนักของโหนด ฉ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 28

0	0	1/48	1/48	1/48	1/48	0	0
0	1/48	1/48	1/48	1/48	1/48	1/48	0
1/48	1/48	1/48	0	0	0	1/48	1/48
0	1/48	1/48	0	0	0	1/48	1/48
1/48	1/48	1/48	0	0	0	1/48	1/48
1/48	1/48	0	0	0	0	1/48	1/48
1/48	1/48	0	0	0	0	1/48	1/48
1/48	1/48	1/48	0	0	0	1/48	1/48
1/48	1/48	0	1/48	0	0	1/48	1/48
0	1/48	1/48	0	0	0	1/48	1/48
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

0	0	1/56	1/56	1/56	1/56	0	0
0	1/56	1/56	1/56	1/56	1/56	1/56	0
1/56	1/56	1/56	0	0	0	1/56	1/56
0	1/56	1/56	0	0	0	1/56	1/56
1/56	1/56	1/56	0	0	0	1/56	1/56
1/56	1/56	0	0	0	0	1/56	1/56
1/56	1/56	0	0	0	0	1/56	1/56
1/56	1/56	1/56	0	0	0	1/56	1/56
1/56	1/56	0	1/56	0	0	1/56	1/56
0	1/56	1/56	0	0	0	1/56	1/56
0	0	0	0	0	0	1/56	1/56
0	0	0	0	0	0	1/56	1/56
0	0	0	0	0	0	1/56	1/56
0	0	0	0	0	0	1/56	1/56
0	0	0	0	0	0	1/56	1/56

28.1) แสดงการกำหนดค่าถ่วงน้ำหนักเริ่มต้นของ ฉ และ ฉุ
โดยใช้ค่าที่ผ่านการนอร์มอล์ไลซ์เพียงอย่างเดียว

0	0	1	1	1	1	0	0
0	1	1	1	1	1	1	0
1	1	1	0	0	0	1	1
0	1	1	0	0	0	1	1
1	1	1	0	0	0	1	1
1	1	0	0	0	0	1	1
1	1	0	0	0	0	1	1
1	1	1	0	0	0	1	1
1	1	0	1	0	0	1	1
0	1	1	0	0	0	1	1
0	0	0	0	0	0	0	1
0	0	0	0	0	0	1	1
0	0	0	0	0	0	1	1
0	0	0	0	0	0	1	0
0	0	0	0	0	0	1	0

28.2) แสดงตัวอย่างอินพุตเวคเตอร์ของ ฉุ ที่เข้าสู่โครงข่าย

แสดงปัญหาการกำหนดค่าถ่วงน้ำหนักเริ่มต้นด้วยวิธีการนำ
ตัวอย่างข้อมูลที่ผ่านการนอร์มอล์ไลซ์มาเป็นต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดค่าติดลบให้กับพื้นของตัวอักษรที่นำมาเป็นต้นแบบของค่าถ่วงน้ำหนักนี้นับเป็นสิ่งที่ยุ่งยากมาก ถ้ากำหนดให้ค่าติดลบมีค่าน้อยเกินก็อาจทำให้ส่วนที่ติดลบไม่สามารถยับยั้งส่วนที่มีค่าเป็นบวกของเนื้อตัวอักษรได้ งานวิจัยนี้จึงกำหนดให้ค่าติดลบมีค่าเท่ากับค่านอร์มอล์ไลซ์ของข้อมูลที่นำมาสอน

ภาพที่ 29 แสดงเมตริกขนาด 8x9 ของค่าถ่วงน้ำหนักเริ่มต้นที่ใช้ในโครงข่าย CPN ของ ก ภาพที่ 29.1 แสดงเมตริกของตัวอย่างข้อมูล ก ที่นำมาเป็นต้นแบบในการกำหนดค่าถ่วงน้ำหนักเริ่มต้นของโครงข่าย ภาพที่ 29.2 แสดงเมตริกของค่าถ่วงน้ำหนักเริ่มต้นของภาพที่ 29.1 ที่ผ่านกระบวนการในข้อ 1.1 และ 1.2

ภาพที่ 29

0	0	1	1	1	1	0	0
0	1	1	1	1	1	1	0
1	1	1	0	0	0	1	1
1	1	1	0	0	0	1	1
0	1	1	0	0	0	1	1
0	1	1	0	0	0	1	1
0	1	1	0	0	0	1	1
0	1	1	0	0	0	1	1
0	1	1	0	0	0	1	1

29.1) แสดงตัวอย่างข้อมูลที่เลือกมาเป็นต้นแบบในการกำหนดค่าถ่วงน้ำหนักเริ่มต้น

-1/40	-1/40	1/40	1/40	1/40	1/40	-1/40	-1/40
-1/40	1/40	1/40	1/40	1/40	1/40	1/40	-1/40
1/40	1/40	1/40	-1/40	-1/40	-1/40	1/40	1/40
1/40	1/40	1/40	-1/40	-1/40	-1/40	1/40	1/40
-1/40	1/40	1/40	-1/40	-1/40	-1/40	1/40	1/40
-1/40	1/40	1/40	-1/40	-1/40	-1/40	1/40	1/40
-1/40	1/40	1/40	-1/40	-1/40	-1/40	1/40	1/40
-1/40	1/40	1/40	-1/40	-1/40	-1/40	1/40	1/40
-1/40	1/40	1/40	-1/40	-1/40	-1/40	1/40	1/40

29.2) แสดงค่าถ่วงน้ำหนักเริ่มต้นของตัวอย่างข้อมูลต้นแบบในภาพที่ 29.1

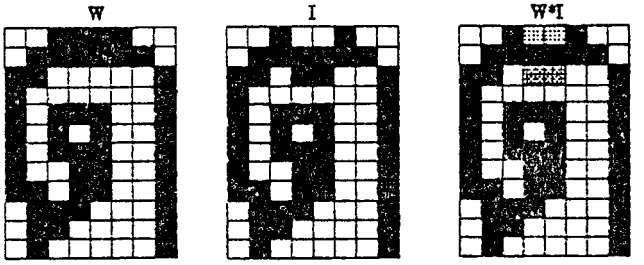
แสดงตัวอย่างเมตริกของค่าถ่วงน้ำหนักเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตัวอย่างปัญหาการสอน ฉ และ ฐ ที่กล่าวข้างต้นเมื่อใช้วิธีการกำหนดค่าถ่วงน้ำหนักเริ่มต้นโดยใช้ตัวอย่างข้อมูลที่ผ่านการนอร์มอล์ไลซ์เป็นต้นแบบและให้พื้นของตัวอักษรที่นำมาเป็นต้นแบบมีค่าติดลบ ซึ่งเป็นผลให้ค่าถ่วงน้ำหนักของโหนด ฐ เป็นผู้ชนะ เนื่องจากส่วนที่เกินของอินพุตเวคเตอร์ ฐ เมื่อนำมาซ้อนทับกับค่าถ่วงน้ำหนักของโหนด ฉ แล้วค่าที่เป็นลบของค่าถ่วงน้ำหนักของโหนด ฉ จะไปยับยั้งส่วนที่เกินของอินพุตเวคเตอร์ ฐ จึงทำให้ค่า net input ของโหนด ฉ มีค่าน้อยกว่าโหนด ฐ (net input ของโหนด ฉ เท่ากับ 0.875 และค่า net input ของโหนด ฐ เท่ากับ 0.894)

2.การกำหนดให้ข้อมูลที่เข้าสู่โครงข่ายอยู่ในรูป +1 และ -1 การกำหนดค่าถ่วงน้ำหนักเริ่มต้นให้มีค่าบวก-ลบและข้อมูลที่เข้าสู่โครงข่ายมีค่า +1 และ 0 ซึ่งค่าที่เป็นลบของค่าถ่วงน้ำหนักช่วยแก้ปัญหา กรณีที่ขนาดเนื้อตัวอักษรของข้อมูลที่เข้าสู่โครงข่ายมีขนาดหนากว่าเนื้อตัวอักษรที่เป็นค่าถ่วงน้ำหนักในโครงข่ายดังเช่นกรณีหัวข้อ 1.2 แต่กรณีที่ขนาดเนื้อตัวอักษรของข้อมูลที่เข้าสู่โครงข่ายมีขนาดเล็กกว่าเนื้อตัวอักษรที่เป็นค่าถ่วงน้ำหนักไม่ได้ถูกนำมาคิดค่า net input ปัญหาที่เห็นได้ชัดคือการแยกแยะระหว่างอินพุตเวคเตอร์ ต กับเวคเตอร์ของค่าถ่วงน้ำหนักของโหนด ด ภาพที่ 30 แสดงการซ้อนทับของอินพุตเวคเตอร์ I ต และเวคเตอร์ของค่าถ่วงน้ำหนัก W ด จากรูปเมื่อมีการคิดค่า net input คือ W*I ซึ่งจากรูปส่วนสีน้ำตาลเข้มแสดงเป็นส่วนยับยั้งของโหนด ด เมื่อมีการใช้ค่าถ่วงน้ำหนักที่มีค่าเป็นลบและอินพุตเวคเตอร์มีค่าเป็น 1 และ 0 ซึ่งจะเห็นว่าส่วนสีน้ำตาลเข้มมีผลเพียงเล็กน้อยในการยับยั้งไม่ให้โหนด ด active ซึ่งในบางครั้งอาจไม่มีผล

ภาพที่ 30



แสดงการซ้อนทับของโหนดของค่าถ่วงน้ำหนัก ด กับข้อมูล ต ที่เข้าสู่โครงข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ถ้ากำหนดให้ข้อมูลที่เข้าสู่โครงข่ายมีค่า +1 และ -1 โดยค่า -1 แทนส่วนที่เป็นพื้นตัวอักษร ซึ่งค่าที่เป็นลบของอินพุตเวคเตอร์นี้เมื่อนำมาเทียบกับค่าถ่วงน้ำหนักที่มีค่าเป็นบวกจะช่วยเพิ่มผลการยับยั้งส่วนเนื้อที่ขาดของข้อมูลที่เข้าสู่โครงข่ายให้มีค่ามากขึ้นอีก จากภาพที่ 30 ภาพของ **W*I** ส่วนของสีเทาอ่อนแสดงส่วนยับยั้งเมื่อมีการใช้ค่าอินพุตเวคเตอร์ที่มีค่าเป็น -1 ซึ่งช่วยเพิ่มผลการยับยั้งไม่ให้โหนดของ ด active มากขึ้นอีก จากตัวอย่างที่แสดงให้เห็นนี้ในงานวิจัยนี้จึงกำหนดให้ข้อมูลที่เข้าสู่โครงข่ายมีค่า +1 และ -1 ซึ่งภาพที่ 31 แสดงตัวอย่างของเมตริกของข้อมูลที่เข้าสู่โครงข่ายของตัวอักษร ก

ภาพที่ 31

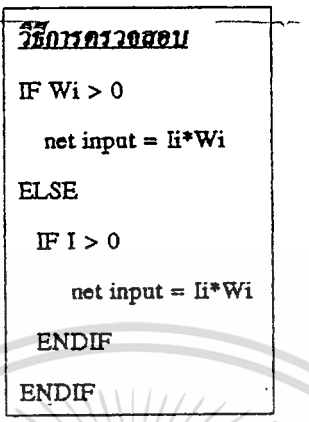
-1	-1	1	1	1	1	-1	-1
-1	1	1	1	1	1	1	-1
1	1	1	-1	-1	-1	1	1
1	1	1	-1	-1	-1	1	1
-1	1	1	-1	-1	-1	1	1
-1	1	1	-1	-1	-1	1	1
-1	1	1	-1	-1	-1	1	1
-1	1	1	-1	-1	-1	1	1
-1	1	1	-1	-1	-1	1	1

แสดงเมตริกของอินพุตเวคเตอร์ของ ก ที่เข้าสู่โครงข่าย

3.วิธีการคิดค่า net input เนื่องจากพื้นของข้อมูลตัวอักษรที่เข้าสู่โครงข่ายและพื้นของค่าถ่วงน้ำหนักมีค่าลบ เพราะฉะนั้นการหาค่า net input จึงไม่นำค่าที่เป็นลบของตัวอักษรที่เข้าสู่โครงข่ายและค่าถ่วงน้ำหนักมาคิดด้วยกันเพราะถือว่าเป็นส่วนของพื้นตัวอักษร ดังนั้นการหาค่า net input จึงต้องมีการตรวจสอบเพื่อไม่ให้มีการนำค่าส่วนที่เป็นลบมาคิด ภาพที่ 32 แสดงขั้นตอนการหาค่า net input โดยมีการตรวจเพื่อไม่ให้มีการคิดค่าส่วนที่เป็นลบของพื้นของตัวอักษรที่เป็นข้อมูลเข้าสู่โครงข่ายและพื้นของค่าถ่วงน้ำหนักที่ถูกกำหนดในขั้นตอนการกำหนดค่าถ่วงน้ำหนักเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 32



แสดงขั้นตอนการคิดค่า net input

เนื่องจากขณะสอนโครงข่ายทำให้ผลรวมของค่าถ่วงน้ำหนักที่มีค่าเป็นบวกของแต่ละโหนดมีค่าไม่เท่ากับหนึ่ง ดังนั้นการคำนวณค่า net input จึงต้องนำค่าผลรวมของค่าถ่วงน้ำหนักที่มีค่าเป็นบวกมาหาร เพื่อให้ค่าผลรวมของค่าถ่วงน้ำหนักมีค่าเป็นหนึ่ง ซึ่งเขียนแทนด้วยสมการที่ 4.2

$$\text{net}_i = \left(\sum_j^n i_j \, w_{ij} \right) / \sum_j^n w_{ij} \quad \text{,for each } w_{ij} > 0 \quad \text{.....(4.2)}$$

โดยที่ net_i : ค่า net input ของชั้นคอมเพททีทีพโหนดที่ i

i_j : ค่าอิลิเมนต์ลำดับที่ j ของอินพุตเวคเตอร์ I

w_{ij} : ค่าอิลิเมนต์ลำดับที่ j ของเวคเตอร์ของค่าถ่วงน้ำหนัก W_i

$\sum_j^n w_{ij}$: ผลรวมของอิลิเมนต์ของค่าถ่วงน้ำหนักที่มีค่าเป็นบวกของ W_i

4.วิธีการปรับค่าถ่วงน้ำหนัก จากที่กล่าวข้างต้นในหัวข้อ 1 และ 2 เรื่องการกำหนดค่าถ่วงน้ำหนักให้อยู่ในรูปของข้อมูลที่ผ่านการนอร์มอล์ไลซ์และกำหนดให้อินพุตเวคเตอร์ที่เป็นข้อมูลเข้าสู่โครงข่ายอยู่ในรูปของค่า +1 และ -1 ดังนั้นขณะสอนโครงข่ายเมื่อมีการปรับค่าถ่วงน้ำหนักของโหนดผู้ชนะจึงต้องทำการนอร์มอล์ไลซ์อินพุตเวคเตอร์ก่อน สมการที่ 4.3 เป็นสมการที่ใช้ในการปรับถ่วงน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\mathbf{W}(t+1) = \mathbf{W}(t) + \alpha\left(\left(1 / \sum_{j=1}^{n} i_j\right) - \mathbf{W}(t)\right) ,$$

$$\sum_{j=1}^{n} i_j \quad \text{where } i_j > 0 \quad \text{.....(4.3)}$$

โดยที่ α : learning rate ซึ่งมีค่าน้อยกว่า 1

$\sum_{j=1}^{n} i_j$: ผลรวมของค่าอิลิเมนต์ของอินพุตเวคเตอร์ที่มีค่ามากกว่า 0

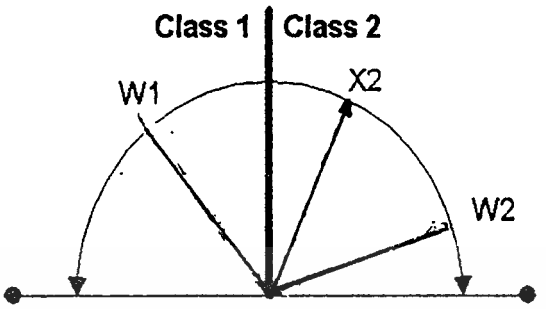
$\mathbf{W}$: เวคเตอร์ของค่าถ่วงน้ำหนักของโหนดผู้ชนะที่เวลา t

การสอนชั้นของคอมเพททิทิพนี้ในช่วงแรกจะกำหนดให้ค่า α มีค่ามากเพื่อให้ทิศทางของค่าถ่วงน้ำหนักปรับเข้าหาข้อมูลที่นำมาสอนได้อย่างรวด ซึ่งทำให้โครงข่ายสามารถจดจำตัวอย่างที่นำมาสอนได้อย่างรวดเร็วเช่นกัน จากนั้นจึงค่อยๆลดค่า α เพื่อป้องกันมิให้ทิศทางของค่าถ่วงน้ำหนักออกนอกบริเวณจุดศูนย์กลางของข้อมูล (centroid region)

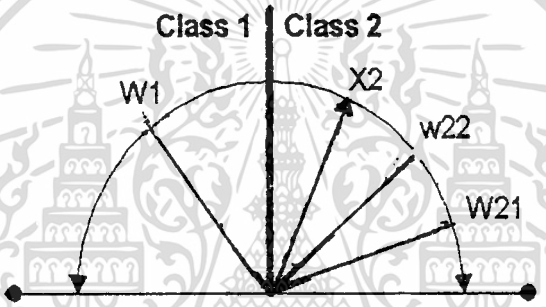
5. การเพิ่มจำนวนโหนดในชั้นคอมเพททิทิพของ CPN จากปัญหาการหาค่าถ่วงน้ำหนักเฉลี่ยที่กล่าวในบทที่ 3 เนื่องจากขอบเขตของกลุ่มข้อมูลกว้างมาก งานวิจัยนี้ใช้วิธีการเพิ่มโหนดของในชั้นคอมเพททิทิพซึ่งการเพิ่มนี้เป็นแบบไดนามิกซ์ (dynamic) เพื่อแบ่งขอบเขตของพื้นที่โหนดผู้ชนะ (win region) ให้มีส่วนย่อยและมีขนาดแคบลง จากปัญหาการหาค่าถ่วงน้ำหนักเฉลี่ยผิดในภาพ 24 แสดงใหม่ในภาพที่ 33.1 ถ้าแบ่งพื้นที่ของโหนดผู้ชนะ W2 ออกเป็น 2 ส่วนย่อยคือ W21 และ W22 แสดงดังภาพที่ 33.2 ซึ่งจะเห็นว่าพื้นที่โหนดผู้ชนะของโหนดทั้งสองมีขนาดแคบมากเมื่อเทียบกับ W2 และเมื่อมีอินพุตเวคเตอร์ X2 เข้าสู่โครงข่ายสามารถแยกแยะได้ว่ากลุ่มข้อมูลของ X2 คือกลุ่มข้อมูลที่ 2 เนื่องจากระยะห่างระหว่างทิศทางของค่าถ่วงน้ำหนัก W21 ของกลุ่มข้อมูลที่ 2 ใกล้กว่า W1 ของกลุ่มข้อมูลที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 33



33.1) แสดงปัญหาการหาค่าถ่วงน้ำหนักผิดในภาพที่ 3.11



33.2) แสดงการแบ่งพื้นที่ขอบเขตผู้ชนะของโหนด W2 ออกเป็น 2 ส่วนคือ W21 และ W22

แสดงการใช้วิธีการเพิ่มโหนดในชั้นคอมเพททิทีพ
เพื่อแก้ปัญหาการหาค่าถ่วงน้ำหนักเฉลี่ยผิดในภาพที่ 24

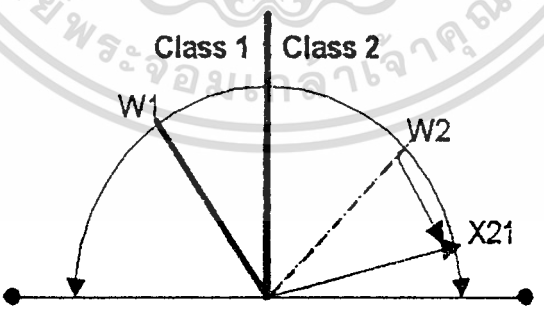
การเพิ่มโหนดเพื่อแบ่งพื้นที่ขอบของโหนดผู้ชนะในงานวิจัยนี้ใช้วิธีการกำหนดค่าเทรสโฮลด์ (Threshold) ซึ่งเป็นเปอร์เซ็นต์ความเหมือนระหว่างอินพุตเวคเตอร์ (input vector) ที่เข้าสู่โครงข่ายกับค่าเวคเตอร์ของค่าถ่วงน้ำหนัก (weight vector) ของโหนดผู้ชนะในชั้นคอมเพททิทีพ โดยที่ถ้าค่าความเหมือนระหว่างอินพุตเวคเตอร์กับค่าเวคเตอร์ของค่าถ่วงน้ำหนักน้อยกว่าเปอร์เซ็นต์ที่กำหนดให้ทำการเพิ่มโหนดใหม่ โดยโหนดที่เพิ่มใหม่นี้จะใช้อินพุตเวคเตอร์ที่เข้าสู่โครงข่ายนี้มาเป็นตัวกำหนดค่าถ่วงน้ำหนักเริ่มต้นโดยใช้วิธีการกำหนดตามหัวข้อ 1 การเปรียบเทียบค่าความเหมือนระหว่างอินพุตเวคเตอร์กับเวคเตอร์ของค่าถ่วงน้ำหนักนี้จะนำค่า net input มาเปรียบเทียบกับค่าเทรสโฮลด์ซึ่งเป็นเปอร์เซ็นต์ความเหมือนระหว่างอินพุตเวคเตอร์กับเวคเตอร์ของค่าถ่วงน้ำหนักที่กำหนดไว้ ถ้ามีค่าน้อยกว่าเปอร์เซ็นต์ที่กำหนดให้ทำการเพิ่มโหนดใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะค่าเทรสโฮลด์ซึ่งเป็นเปอร์เซ็นต์ความเหมือนระหว่างอินพุตเวคเตอร์กับเวคเตอร์ของค่าถ่วงน้ำหนักของแต่ละกลุ่มข้อมูลมีค่าไม่เท่ากันขึ้นอยู่กับชนิดของกลุ่มข้อมูล (ในงานวิจัยนี้กลุ่มข้อมูล หมายถึง ข้อมูลตัวอักษร ก,ข เป็นต้น) ดังนั้นการกำหนดค่าเทรสโฮลด์จึงต้องการระมัดระวัง ทั้งนี้เนื่องจากการกำหนดค่าเทรสโฮลด์ให้มีค่าสูงๆจะทำให้จำนวนโหนดในชั้นคอมเพททิทิพมีการเพิ่มเป็นจำนวนมาก เพราะการกำหนดค่าเทรสโฮลด์ให้มีค่าสูงๆหมายถึง การกำหนดให้ลักษณะของอินพุตเวคเตอร์ที่มาตกในโหนดผู้ชนะนี้มีลักษณะเหมือนกับเวคเตอร์ของค่าถ่วงน้ำหนักมากจึงจะยอมให้มีการปรับค่าถ่วงน้ำหนัก และถ้าลักษณะของอินพุตเวคเตอร์มีความเหมือนน้อยกว่าค่าเทรสโฮลด์ที่กำหนดจะทำการเพิ่มโหนดในชั้นคอมเพททิทิพ โดยใช้ค่าของอินพุตเวคเตอร์ที่มาตกในโหนดผู้ชนะนี้เป็นตัวกำหนดค่าถ่วงน้ำหนักเริ่มต้นของโหนดใหม่ที่เพิ่มขึ้น

นอกจากการแก้ปัญหาการหาค่าเฉลี่ยของค่าถ่วงน้ำหนักผิดแล้ว วิธีนี้ยังช่วยแก้ปัญหาเรียนผิดโหนดกรณีระยะห่างระหว่างขอบเขตของตัวอย่างข้อมูลแต่ละกลุ่มมีระยะใกล้กัน ซึ่งช่วยให้โครงข่ายเรียนรู้ตัวอย่างที่นำมาสอนได้อย่างรวดเร็วขึ้น จากปัญหาภาพที่ 22.1 นำมาแสดงใหม่ในภาพที่ 34.1 ถ้าเราเพิ่มโหนดของ W22 เมื่อมีการสอนตัวอย่างข้อมูล X21 แทนการปรับค่าถ่วงน้ำหนักซึ่งแสดงดังภาพที่ 34.2 จะแก้ปัญหาการเรียนผิดโหนดในปัญหาภาพที่ 22 ได้ กล่าวคือ เมื่อมีข้อมูล X22 เข้าสู่โครงข่ายทิศทางของค่าถ่วงน้ำหนักในกลุ่มข้อมูลที่ 2 คือ W2 ยังคงมีระยะใกล้กับ X22 มากที่สุดแสดงดังภาพที่ 34.3 เนื่องจากโหนด W22 ที่เพิ่มในขณะสอนตัวอย่าง X21

ภาพที่ 34

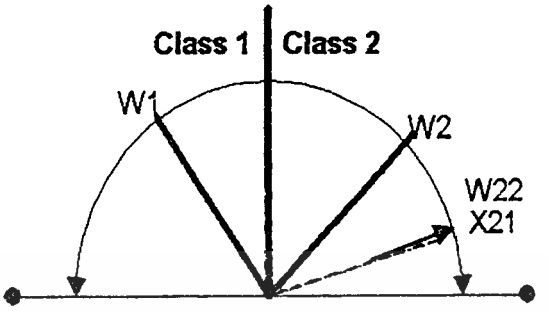


34.1) แสดงทิศทางการปรับทิศทางของเวคเตอร์ของค่าถ่วงน้ำหนัก W2 เมื่อมีอินพุตเวคเตอร์ X21 เข้าสู่โครงข่ายของภาพที่ 22.1 อีกครั้ง

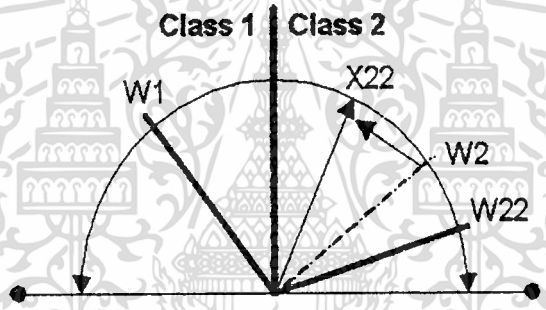
แสดงการแก้ปัญหาภาพที่ 22 ด้วยวิธีการเพิ่มโหนดในชั้นคอมเพททิทิพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 34 (ต่อ)



34.2) แสดงการเพิ่มโหนด W22 ในชั้นคอมเพททิทิพ เมื่อมีอินพุต
เวคเตอร์ X21 เข้าสู่โครงข่ายแทนการปรับค่าถ่วงน้ำหนักของ W2



34.3) แสดงปรับค่าถ่วงน้ำหนัก W2 เมื่อมีอินพุตเวคเตอร์ X22
เข้าสู่โครงข่ายหลังการเพิ่มโหนด W22 ในชั้นคอมเพททิทิพ

แสดงการแก้ปัญหาภาพที่ 22 ด้วยวิธีการเพิ่มโหนดในชั้นคอมเพททิทิพ

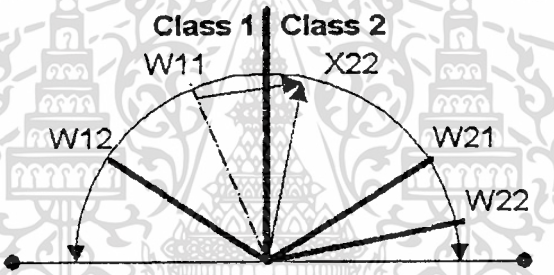
6. การแยกสอน จากปัญหาการเรียนผิดโหนดในบทที่ 3 กรณีระยะห่างระหว่างขอบเขตของตัวอย่างข้อมูลในกลุ่มต่างๆมีระยะใกล้กัน สามารถแก้ปัญหาได้ด้วยวิธีการเพิ่มโหนดในชั้นคอมเพททิทิพ เนื่องจากระยะห่างของทิศทางระหว่างค่าถ่วงน้ำหนักก่อนการนำอินพุตเวคเตอร์มาสอนโครงข่ายมีระยะใกล้กัน

ในตัวอย่างภาพที่ 22 จะสังเกตเห็นว่าทิศทางของ X21 ซึ่งเป็นข้อมูลของกลุ่มข้อมูลที่ 2 ก่อนนำเข้ามาสอนโครงข่ายนั้นมีระยะห่างใกล้กับค่าถ่วงน้ำหนัก W2 ซึ่งเป็นค่าถ่วงน้ำหนักของกลุ่มข้อมูลที่ 2 เช่นกัน กรณีนี้สามารถใช้วิธีการเพิ่มโหนดในการสอน เพราะเมื่อนำ X21 เข้ามาสอนโครงข่ายโหนดของ W2 จะเป็นผู้ชนะ

เอกสารนี้มีลิขสิทธิ์ สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่จากภาพที่ 34.3 ซึ่งเป็นการแก้ปัญหาในตัวอย่างภาพที่ 22 ด้วยการเพิ่มโหนดในชั้นคอมเพททิทิพ ถ้าข้อมูล X22 ของกลุ่มข้อมูลที่ 2 ก่อนนำเข้ามาสอนโครงข่ายมีระยะไกลจาก W2 และ W22 ซึ่งเป็นค่าถ่วงน้ำหนักของกลุ่มที่ 2 เมื่อเทียบกับ W11 ของกลุ่มข้อมูลที่ 1 แล้ว ซึ่งแสดงดังภาพที่ 35 เมื่อนำ X22 เข้ามาสอนโครงข่ายค่าถ่วงน้ำหนัก W11 จะถูกเรียนแทนถึงแม้ว่าจะมีการเพิ่มโหนด W22 ในการสอนข้อมูล X21 แล้วก็ไม่สามารถแก้ปัญหานี้ได้ ทั้งนี้เนื่องจากทิศของข้อมูลที่นำมาสอนโครงข่ายมีระยะไกลจากค่าถ่วงน้ำหนักของกลุ่มข้อมูลที่ต้องการสอนดังนั้นจึงทำให้เกิดการเรียนผิดโหนดขึ้น

ภาพที่ 35

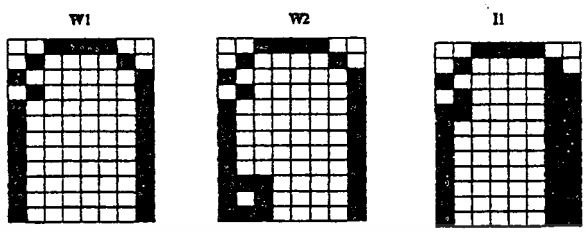


แสดงปัญหาเรียนผิดโหนดกรณีระยะห่างของอินพุตเวคเตอร์ X22 ก่อนการสอน มีระยะห่างน้องกว่าค่าถ่วงน้ำหนักของกลุ่มข้อมูลที่ 2 ซึ่งเป็นผลให้ W11 ถูกเรียน

จากปัญหาที่กล่าวในภาพที่ 35 สามารถแสดงตัวอย่างของปัญหานี้ด้วยขั้นตอนการสอนตัวอักษรภาษาไทยได้ดังนี้ ภาพที่ 36.1 W1,W2 คือค่าถ่วงน้ำหนักที่ใช้แยกตัวอักษร ก และ ฉ ซึ่งลักษณะตัวบาง เมื่ออินพุตเวคเตอร์ I1 (ก) ที่มีลักษณะหนาเข้ามาสอนโครงข่าย W1 จะถูกสอนดังแสดงในภาพที่ 36.2 จากนั้นนำอินพุตเวคเตอร์ I2 (ฉ) ที่มีลักษณะหนาเช่นเดียวกับ I1 เข้ามาสอน ซึ่งจากภาพที่ 36.2 จะเห็นว่า W1 เหมือนกับ I2 มากกว่า W2 เนื่องจากความหนาของ W1 หลังการสอน I1 ทำให้ I2 ไปตกที่ W1 ซึ่งผิดโหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 36



36.1) แสดงการลักษณะของค่าถ่วงน้ำหนัก W1 (ก),W2 (ถ)
และอินพุตเวคเตอร์ I1 (ก) ก่อนการสอนโครงข่าย

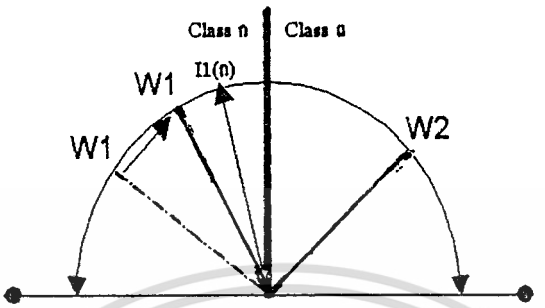
36.2) แสดงการลักษณะของค่าถ่วงน้ำหนัก W1 (ก) เมื่อได้รับการสอน I1
และอินพุตเวคเตอร์ I2 (ถ) ก่อนการสอนโครงข่าย

แสดงตัวอย่างอย่างปัญหาการสอน ก,ถ กรณีทิศของข้อมูลที่นำมาสอน
โครงข่ายมีระยะไกลจากค่าถ่วงน้ำหนักของกลุ่มข้อมูลที่ต้องการสอน

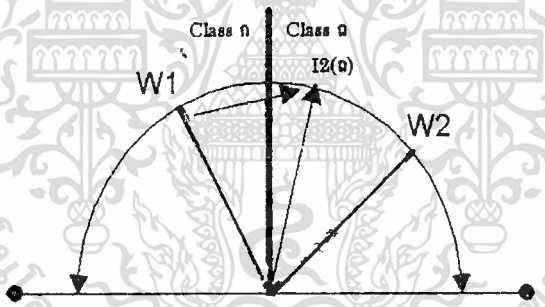
ปัญหาภาพที่ 36 สามารถอธิบายด้วยทิศทางการปรับค่าถ่วงน้ำหนักดังภาพที่ 37 ภาพที่ 37.1 แสดงทิศทางค่าถ่วงน้ำหนักเริ่มต้นของ W1,W2 และทิศการปรับของ W1 หลังจากนำ I1 เข้ามาสอนโครงข่าย จากนั้นเมื่อนำ I2 เข้ามาสอนในโครงข่ายซึ่งแสดงดังภาพที่ 37.2 จากรูปจะเห็นว่าไม่มีทิศทางใดในกลุ่ม ถ อยู่ใกล้กับ I2 เลยเมื่อเทียบกับกลุ่ม ก ดังนั้นค่าถ่วงน้ำหนักของกลุ่ม ก จึงได้รับการสอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 37



37.1) แสดงทิศทางการปรับค่าถ่วงน้ำหนัก W1 ของภาพที่ 36.2
เมื่อมีอินพุตเวคเตอร์ I1 (n) เข้าสู่โครงข่าย



37.2) แสดงทิศทางการปรับค่าถ่วงน้ำหนัก W1 ของภาพที่ 36.2
เมื่อมีอินพุตเวคเตอร์ I2 (n) เข้าสู่โครงข่าย

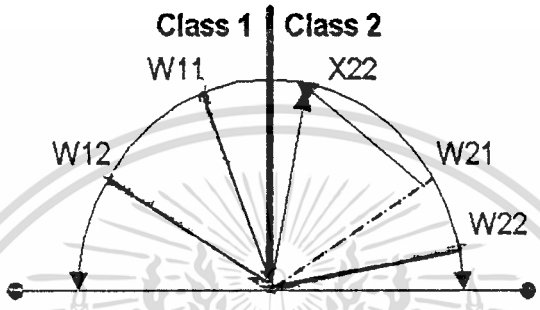
แสดงอธิบายปัญหาภาพที่ 36 ด้วยรูปทิศทางการปรับค่าถ่วงน้ำหนัก

เนื่องจากในการสอน CPN เพื่อให้แยกแยะตัวอักษรภาษาไทยนั้น เราได้จัดแบ่งข้อมูลที่นำมาสอนเป็นกลุ่มที่แน่นอนอยู่แล้วและลักษณะการเรียนของ CPN จะคัดเลือกโหนดผู้ชนะ เพียงโหนดเดียวเพื่อนำมาใช้ในการเรียนข้อมูลที่เข้าสู่โครงข่ายหรือปรับค่าถ่วงน้ำหนัก งานวิจัยนี้จึงใช้วิธีการแยกสอนตัวอย่างข้อมูลที่ละกลุ่ม เพื่อบังคับให้ข้อมูลที่นำมาสอนโครงข่ายตกอยู่ในขอบเขตพื้นที่ของโหนดผู้ชนะ (win region) ของกลุ่มข้อมูลต้องการสอนได้ ซึ่งวิธีจะป้องกันปัญหาการ

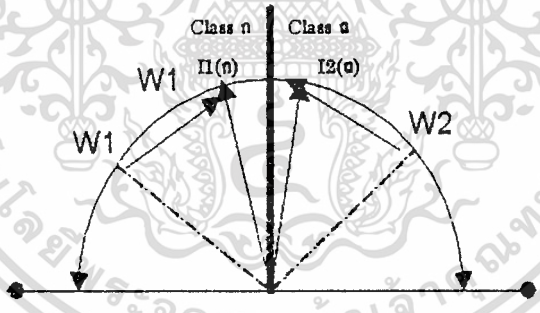
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียนผิดโหนดกรณีที่กล่าวข้างต้น ปัญหาจากภาพที่ 36 และ 37 สามารถแสดงทิศทางการปรับค่าถ่วงน้ำหนักด้วยวิธีการแยกสอนด้วยภาพที่ 38

ภาพที่ 38



38.1) แสดงทิศทางการปรับค่าถ่วงน้ำหนักของภาพที่ 36 กรณีแยกสอน



38.2) แสดงทิศทางการปรับค่าถ่วงน้ำหนักของภาพที่ 37 กรณีแยกสอน

แสดงการทิศทางการปรับค่าถ่วงน้ำหนักของภาพที่ 36 และ 37 กรณีแยกสอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปขั้นตอนการสอนโครงข่าย

1. จัดแบ่งกลุ่มตัวอย่างที่นำมาสอนออกเป็นกลุ่ม (ก,ข,ค...) และ
   กำหนดค่าเทรสโฮลด์ซึ่งเป็นเปอร์เซ็นต์ความเหมือนของอินพุตเวคเตอร์กับค่าเวคเตอร์
   ของค่าถ่วงน้ำหนักให้กับแต่ละข้อมูลที่นำมาสอน
2. กำหนดค่า α
3. ทำการแยกสอนกลุ่มตัวอย่างข้อมูลที่นำมาสอนทีละกลุ่ม
   3.1. เลือกตัวอย่างอินพุตเวคเตอร์มาหนึ่งตัวเพื่อทำการกำหนด
   ค่าถ่วงน้ำหนักเริ่มต้นตามหัวข้อ 1
   3.2. สุ่มเลือกตัวอย่างอินพุตเวคเตอร์เพื่อนำมาสอน แปลงค่า 0,1 เป็น -1,1
   3.3. หาค่า net input ตามหัวข้อ 3 เลือกโหนดที่มีค่า net input มากที่สุดเป็นโหนดผู้ชนะ
   3.4. เปรียบเทียบค่า net input ของโหนดผู้ชนะกับค่าเทรสโฮลด์ซึ่งเป็นเปอร์เซ็นต์ความเหมือน
   ระหว่างอินพุตเวคเตอร์กับค่าเวคเตอร์ของค่าถ่วงน้ำหนักของโหนดผลชนะนั้น
   ถ้า ค่า net input น้อยกว่าค่าเปอร์เซ็นต์เทรสโฮลด์ที่กำหนด
   3.4.1. เพิ่มโหนดในชั้นคอมเพททิทิพตามหัวข้อ 1
   ไม่ใช่
   3.4.2. ทำการปรับค่าถ่วงน้ำหนักตามหัวข้อ 4
   3.5. ทำให้ได้ตามจำนวนรอบที่ต้องการ (จำนวนรอบในการสอนข้อมูลแต่ละกลุ่มควรเท่ากัน)
4. ทำการทดสอบรวม
   ถ้า เปอร์เซ็นต์ความถูกต้องของทดสอบรวมได้น้อยกว่าที่กำหนด
   4.1. เพิ่มค่าเทรสโฮลด์ซึ่งเป็นเปอร์เซ็นต์ความเหมือนของอินพุตเวคเตอร์
   กับค่าเวคเตอร์ของค่าถ่วงน้ำหนัก (สำดับการสอนตัวอักษรเหมือนเดิม)
   4.2. ย้อนกลับเริ่มต้นไปสอนข้อมูลกลุ่มนี้ใหม่อีกครั้ง
   ไม่ใช้
   4.3. ให้ทำการลดค่า α ลงกลับทำข้อ 3 เพื่อสอนข้อมูลในรอบถัดไป
5. ทำจนกระทั่งได้ค่าเปอร์เซ็นต์ความถูกต้องในการแยกแยะตัวอย่างที่นำมาสอนตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

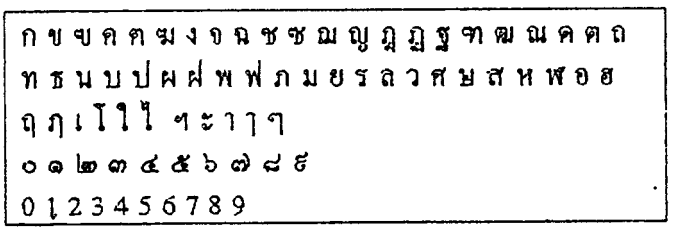
ผลการทดลองการแยกแยะตัวพิมพ์อักษรภาษาไทย

งานวิจัยนี้ได้ทดลองการแยกแยะตัวอักษรโดยใช้ตัวพิมพ์อักษรภาษาไทย โดยรูปแบบ (font) ตัวอักษรที่นำมาใช้ในทดลองคือ AngsanaUPC ขนาด 12 point ซึ่งมีความละเอียดที่ใช้ในการสแกน คือ 300 จุดต่อนิ้วและอัตราส่วนการย่อขยายคือ 100 เปอร์เซ็นต์ การทดลองแบ่งออกเป็น 2 ส่วนคือ การทดลองการแยกแยะตัวอักษรที่ละชุดของตัวอักษรและการทดสอบการแยกแยะตัวอักษรที่อยู่ในรูปของประโยค

<u>การทดลองการแยกแยะตัวอักษรทีละชุดของตัวอักษร</u>

การทดลองนี้มีจุดประสงค์เพื่อแสดงผลของการเพิ่มโหนดในชั้นคอมเพททิทิพในการสอนโครงข่ายเพื่อการแยกแยะตัวพิมพ์อักษรภาษาไทย ตัวอักษรที่นำมาทดลองเป็นตัวพิมพ์อักษรภาษาไทยจำนวน 75 ตัว มาใช้ในการทดสอบ โดยตัวอักษรที่นำมาทดสอบเป็นตัวอักษรที่อยู่ในระดับบรรทัดกลางของระดับภาษาไทยซึ่งประกอบด้วย พยัญชนะจำนวน 44 ตัว,เลขไทยและเลขอารบิคจำนวน 20 ตัวและสระต่างๆ แสดงดังภาพที่ 39

ภาพที่ 39

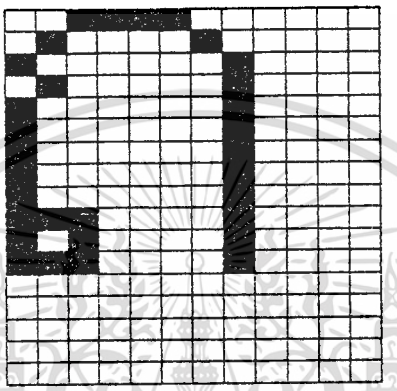


แสดงตัวอักษรที่นำมาทดลองการแยกแยะทีละชุดตัวอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขนาดของเมตริกที่เป็นข้อมูลเข้าสู่โครงข่ายมีขนาด 28 x 38 ซึ่งตำแหน่งการวางภาพตัวอักษรบนเมตริกคืออยู่ที่ตำแหน่งชิดมุมบนซ้ายของเมตริกที่เป็นข้อมูลเข้าสู่โครงข่าย ภาพที่ 40 แสดงตัวอย่างการวางภาพตัวอักษรบนเมตริกขนาด 12x16 เพื่อเป็นข้อมูลที่เข้าสู่โครงข่าย

ภาพที่ 40



แสดงการวางตัวอักษรบนเมตริกเพื่อเป็นข้อมูลเข้าสู่โครงข่ายที่ใช้ในการทดลอง

ตัวอย่างของตัวอักษรที่นำมาใช้ในการทดลองประกอบด้วยตัวอักษร 4 ชุด ซึ่งสแกนที่ความสว่างต่างๆกัน คือ

1.พิมพ์ผ่านเครื่องเลเซอร์ ที่ความสว่าง 100 เปอร์เซ็นต์

2.พิมพ์ผ่านเครื่องเลเซอร์ ที่ความสว่าง 50 เปอร์เซ็นต์

3.พิมพ์ผ่านเครื่องเลเซอร์แล้วผ่านการถ่ายเอกสาร ที่ความสว่าง 100 เปอร์เซ็นต์

4.พิมพ์ผ่านเครื่องเลเซอร์แล้วผ่านการถ่ายเอกสาร ที่ความสว่าง 50 เปอร์เซ็นต์

โดยแต่ละชุดมีจำนวน 15,000 ตัวอักษรเพื่อนำมาใช้ในการทดสอบการแยกตัวอักษร ซึ่งแบ่งจำนวนตัวอักษรในแต่ละชุดออกเป็นจำนวน 200 ตัวต่อกลุ่มตัวอักษรซึ่งมี 75 กลุ่มตัวอักษร และคุณภาพของภาพตัวอักษรที่ผ่านการสแกนด้วยความสว่าง 100 เปอร์เซ็นต์จะมีความชัดของภาพมากกว่า 50 เปอร์เซ็นต์

การทดลองในงานวิจัยนี้ได้สร้างโครงข่ายขึ้นมาเพื่อทำการทดสอบ 3 โครงข่าย ซึ่งขอเรียกโครงข่ายทั้ง 3 ชุดนี้ว่า โครงข่าย A,B และ C ตามลำดับ ตัวอย่างภาพตัวอักษรที่เลือกนำมาเป็นแบบฝึกสอนให้กับโครงข่ายทั้ง 3 นี้เป็นชุดเดียวกัน โดยคัดเลือกจากกลุ่มของตัวอักษรที่นำมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทดสอบกลุ่มตัวอักษรละ 20 แบบ มีจำนวนรวมทั้งหมด 1500 ตัวอักษร และอุปกรณ์คอมพิวเตอร์ที่ใช้ในการฝึกสอนคือ เครื่อง HP 9000/827

การสอนโครงข่ายทั้ง 3 ชุดนี้ขั้นตอนแรกทำการกำหนดค่าถ่วงน้ำหนักเริ่มต้น โดยการคัดเลือกจากชุดตัวอักษรที่นำมาเป็นครูฝึกสอนโครงข่ายมาเป็นต้นแบบของค่าถ่วงน้ำหนัก ซึ่งตัวอักษรที่นำมาเป็นต้นแบบของค่าถ่วงน้ำหนักเริ่มต้นของทั้ง 3 โครงข่ายเป็นชุดเดียวกัน จำนวนโหนดของค่าถ่วงน้ำหนักเริ่มต้นในชั้นคอมเพทพิพิทเท่ากับจำนวนกลุ่มตัวอักษรที่นำมาใช้สอนคือ 75 โหนด การสอนนี้กำหนดค่า α เริ่มต้นเท่ากับ 0.8 จากนั้นทำการลดค่าลงครั้งละ 0.2 และจำนวนรอบของการสอนโครงข่ายจะทำกระทั่งสามารถแยกแยะตัวอย่างที่นำมาสอนได้ 100 เปอร์เซ็นต์

วิธีการทดสอบผลการทดลองนี้ จะนำตัวอักษรที่ผ่านการเซกเมนต์แล้วทีละกลุ่มตัวอักษรผ่านเข้าสู่โครงข่ายที่ได้รับการสอนตัวอย่างจนเสร็จสิ้นแล้ว จากนั้นนับจำนวนตัวอักษรที่สามารถวิเคราะห์หรือแยกแยะได้ ซึ่งผลที่ได้จากทดลองนี้จะไม่มีการคิดความผิดพลาดที่เกิดจากการเซกเมนต์ตัวอักษร

การสอนโครงข่ายทั้ง 3 ชุดนี้มีความแตกต่างอยู่ที่ค่าของเทรสโฮลด์ ซึ่งเป็นเปอร์เซ็นต์ของความเหมือนของอินพุตเวคเตอร์และเวคเตอร์ของค่าถ่วงน้ำหนักในชั้นคอมเพททิทิพที่กำหนดให้กับแต่ละโครงข่าย วิธีการกำหนดค่าเทรสโฮลด์ของโครงข่ายชุด B และ C จะนำค่าเทรสโฮลด์ที่ได้จากการสอนโครงข่าย A มาทำการแก้ไข โดยพิจารณาจากผลการทดลองของโครงข่าย A เพื่อให้ผลการแยกแยะตัวดีขึ้น

ตารางที่ 3 แสดงค่าเทรสโฮลด์ที่ใช้ในการสอนโครงข่ายชุด A และตารางที่ 4 แสดงเปอร์เซ็นต์การเรียนรู้ของโครงข่ายทุกๆ 4 รอบจากจำนวนตัวอย่าง 1,500 ตัว จากตารางที่ 4 จำนวนโหนดที่ได้จากสอนของชั้นคอมเพททิทิพมีทั้งหมด 141 โหนด

จากที่กล่าวข้างต้นในบทที่ 4 ในหัวข้อเรื่องการเพิ่มโหนดในชั้นคอมเพททิทิพ ว่าการกำหนดค่าเทรสโฮลด์ให้กับแต่ละกลุ่มข้อมูลตัวอักษรนั้นมีค่าไม่เท่ากัน ตารางที่ 3 จะสังเกตเห็นว่าค่าเทรสโฮลด์ที่มีค่ามากจะอยู่ในกลุ่มของตัวอักษรที่มีลักษณะคล้ายกันเช่น ด,ต,ถ,ศ เนื่องจากของเขตของกลุ่มข้อมูลมีระยะใกล้เคียงกัน ดังนั้นจึงต้องการแบ่งพื้นที่ของโหนดผู้ชนะให้มีขนาดแคบ ส่วนกลุ่มตัวอักษรอื่นๆ เช่น ฐ,จ,ง มีค่าน้อยเนื่องจากไม่มีรูปแบบใดมีลักษณะใกล้เคียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3

ตัวอักษร	ค่า Threshold	ตัวอักษร	ค่า Threshold	ตัวอักษร	ค่า Threshold	ตัวอักษร	ค่า Threshold
ก	45	ข	30	ฃ	10	ค	60
ฅ	55	ฆ	10	ง	10	จ	30
ฉ	10	ช	25	ซ	25	ฌ	40
ญ	20	ฎ	20	ฏ	25	ฐ	10
ฑ	10	ฒ	15	ณ	40	ด	60
ต	60	ถ	45	ท	15	ธ	10
น	10	บ	10	ป	10	ผ	10
ฝ	10	พ	50	ฟ	10	ภ	40
ม	35	ย	35	ร	10	ฤ	50
ล	40	ฦ	50	ว	10	ศ	10
ษ	25	ส	20	ห	10	พ	10
อ	35	ฮ	10	ฯ	10	ะ	5
า	10	0	30	1	10	2	5
3	10	4	10	5	5	6	10
7	10	8	10	9	10	เ	10
โ	35	ไ	65	ใ	60	ๆ	15
ๆ	10	๐	10	๑	30	๒	10
๓	10	๔	10	๕	10	๖	10
๗	10	๘	10	๙	10		

แสดงค่าเทรสโฮลด์ที่ใช้ในการสอนโครงข่าย A

ตารางที่ 5 แสดงจำนวนโหนดในชั้นคอมเพททิทิพของแต่ละกลุ่มข้อมูลตัวอักษร โหนดของกลุ่มที่มีค่าเทรสโฮลด์ที่มีค่ามากจะมีโหนดมาก เช่น ค 5 โหนด, ต 5 โหนด, ด 4 โหนด, ต 7 โหนด, ภ 4 โหนด, ฤ 4 โหนด เป็นต้น ส่วนโหนดที่มีค่าเทรสโฮลด์น้อยจำนวนโหนดก็จะน้อย เช่น ข 1 โหนด, ง 1 โหนด, จ 2 โหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4

ครั้ง	ค่า α	จำนวนโหนด	จำนวนตัวอย่างที่ถูกต้อง	% ความถูกต้องของการสอนตัวอย่าง
0	-	75	-	-
1	0.8	123	1454	96.93
2	0.8	130	1469	97.93
3	0.8	139	1473	98.19
4	0.6	139	1488	99.19
5	0.6	139	1492	99.46
6	0.6	140	1482	98.79
7	0.6	141	1488	99.19
8	0.4	141	1496	99.73
9	0.4	141	1499	99.93
10	0.4	141	1495	99.66
11	0.4	141	1498	99.86
12	0.2	141	1500	100
13	0.2	141	1500	100
14	0.2	141	1500	100
15	0.2	141	1500	100

เปอร์เซ็นต์การเรียนทุก ๆ 4 รอบ ใช้จำนวนข้อมูล
ที่นำมาสอน 1500 ตัวของโครงข่าย A

ตารางที่ 6 แสดงผลการทดสอบการแยกแยะตัวพิมพ์อักษรภาษาไทยด้วยโครงข่าย A จากผลการทดลองเปอร์เซ็นต์ของตัวอักษรที่พิมพ์ผ่านเครื่องเลเซอร์แล้วสแกนด้วยความสว่าง 100 เปอร์เซ็นต์มีความถูกต้องมากที่สุด เนื่องจากคุณภาพของตัวอักษรที่ได้จากการสแกนมีความชัดมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5

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

แสดงจำนวนโหนดในชั้นคอมเพททิทิพของแต่ละกลุ่มตัวอักษรของโครงข่าย A

ตารางที่ 6

ชนิดกระดาษที่ทดสอบ (%ความสว่าง)	จำนวนตัวอักษรที่ผิด	เปอร์เซ็นต์ความถูก
พิมพ์ laser (100 %)	27	99.48
พิมพ์ laser (50 %)	697	95.35
ถ่ายเอกสาร (100%)	92	99.38
ถ่ายเอกสาร (50%)	959	93.60

แสดงผลการทดสอบการแยกตัวอักษรภาษาไทยชุดละ 15,000 ตัว โดยใช้โครงข่าย A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 7

ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด
ก	-	ข	ข (35)	ฃ	-	ค	ค (4)
ฅ	ค (4)	ฆ	-	ง	-	จ	-
ฉ	-	ช	ซ (1)	ซ	-	ฌ	-
ญ	-	ฎ	ฎ (5)	ฏ	-	ฐ	-
ฑ	-	ฒ	-	ณ	-	ด	ต (2)
ต	-	ถ	-	ท	-	ธ	-
น	-	บ	-	ป	-	ผ	-
ฝ	-	พ	-	ฟ	-	ภ	-
ม	-	ย	-	ร	-	ฤ	-
ล	-	ฦ	-	ว	-	ศ	-
ษ	-	ส	-	ห	พ (1)	พ	-
อ	-	ฮ	-	ฯ	-	ะ	-
า	-	0	-	1	-	2	-
3	-	4	-	5	-	6	-
7	-	8	-	9	-	เ	-
โ	ไ (2)	ไ	โ(5) ,ไ (5)	ใ	ไ (13)	ๅ	-
ๆ	-	๐	-	๑	-	๒	-
๓	-	๔	-	๕	-	๖	-
๗	-	๘	-	๙	-		

แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษพิมพ์เลเซอร์ ที่ความสว่าง 100 เปอร์เซ็นต์ของโครงข่าย A

ตารางที่ 7 ถึง 10 แสดงจำนวนตัวอักษรที่แยกแยะผิดของแต่ละกลุ่มของโครงข่าย A โดยเลขในวงเล็บแสดงจำนวนตัวอักษรที่ผิด และตัวอักษรข้างหน้าวงเล็บแสดงผลลัพธ์ของกลุ่มที่โครงข่ายแยะแยกได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 8

ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด
ก	17*	ช	ช(81)	ซ	พ(2)	ค	57*
ค	29*	ฌ	26*	ง	-	จ	ฉ(8)
ฉ	-	ร	ซ(45)	ซ	-	ฌ	-
ญ	-	ฎ	-	ฏ	ฎ(59)	ฐ	-
ท	ฒ(1)	ฒ	-	ณ	ณ(30)	ค	ต(23)
ต	-	ถ	-	ท	-	ธ	บ(1),จ(1)
น	บ(1)	บ	-	ป	-	ผ	-
ฝ	-	พ	ท(53)	ฟ	-	ภ	ภ(4)
ม	บ(1)	ย	อ(1)	ร	ธ(37)	ฤ	-
ล	-	ฦ	-	ว	-	ศ	ต(1),ส(20),พ(2)
ษ	-	ส	-	ห	ท(47),พ(18)	พ	-
อ	-	ฮ	-	ฯ	-	ะ	-
า	ว(35)	0	-	1	-	2	-
3	-	4	ย(3)	5	-	6	1(12)
7	-	8	-	9	-	เ	-
โ	ใ(2),ไ(1)	ใ	ไ(37)	ไ	-	ๅ	-
ๆ	-	๐	-	๑	-	๒	-
๓	-	๔	-	๕	-	๖	-
๗	-	๘	๗(1)	๙	-		

8.1) แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มตัวอักษร (*ดูตารางที่ 8.2)

ตัวอักษร	ผลลัพธ์*	ตัวอักษร	ผลลัพธ์*
ก	ก(2),ถ(8),ท(1),ภ(4),ศ(2)	ค	ค(16),ค(24),ต(13),ศ(4)
ต	ก(2),ค(3),ต(24)	ฌ	น(8),ฌ(15),พ(3)

8.2) แสดงตัวอย่างผลลัพธ์ที่นำไปทดสอบการแยกตัวอักษรจากตารางที่ 8.1

แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษพิมพ์เลเซอร์

ที่ความสว่าง 50 เปอร์เซ็นต์ของโครงข่าย A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 9

ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด
ก	-	ช	ช(4)	ซ	ซ(1)	ค	ค(4)
ต	ค(4)	ฌ	-	ง	-	จ	ฮ(1)
ฉ	น(1)	ร	บ(1),ช(1)	ช	-	ฌ	-
ญ	-	ฎ	ฎ(2)	ฎ	-	ฐ	-
ฑ	-	ฒ	-	ฌ	-	ด	ต(1)
ต	ค(1),ร(1)	ถ	ท(1)	ท	พ(1)	ธ	-
น	-	บ	-	ป	ฟ(2)	ผ	-
ฝ	-	พ	-	ฟ	-	ภ	-
ม	-	ย	ค(1)	ร	-	ฤ	ถ(1)
ล	ท(1)	ภ	ภ(6)	ว	-	ศ	ส(5)
ษ	-	ส	บ(1)	ห	-	พ	พ(1)
อ	-	ฮ	-	ฯ	-	ะ	ฮ(1)
า	ว(4)	0	-	1	-	2	-
3	า(1)	4	-	5	-	6	-
7	-	8	-	9	-	เ	-
โ	-	ใ	โ(8),ไ(21)	ไ	ใ(4)	ๆ	-
ๆ	-	๐	บ(1)	๑	-	๒	-
๓	-	๔	-	๕	-	๖	๐(1),ฮ(1)
๗	-	๘	-	๘	ฮ(1)		

แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษถ่ายเอกสาร
ที่ความสว่าง 100 เปอร์เซ็นต์ของโครงข่าย A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 10

ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด
ก	8*	ช	ช(44),ช(4)	ข	-	ค	71*
ค	ค(8),ค(3),ค(25)	ฆ	ฑ(1),ฌ(7)	ง	-	จ	ฉ(2)
ฉ	พ(1)	ช	ช(32)	ช	พ(31)	ฌ	-
ญ	-	ฎ	-	ฏ	ฎ(37)	ฐ	-
ฑ	ฒ(2)	ฒ	-	ณ	ฌ(41)	ค	ค(38)
ต	-	ถ	-	ท	พ(53)	ธ	ข(6).ฉ(2)
น	ท(3),บ(3)	บ	-	ป	-	ผ	-
ฝ	ฟ(1)	พ	-	ฟ	ป(1)	ภ	ฦ(7)
ม	บ(8),ษ(1)	ย	ร(1),ฮ(1)	ร	ธ(44)	ฤ	ถ(3)
ล	-	ฦ	-	ว	ช(2),ร(1)	ศ	39*
ษ	ธ(1),พ(6)	ส	-	ห	92*	พ	พ(15)
อ	ค(9)	ฮ	-	ฯ	ศ(1)	ะ	ฮ(1)
า	ธ(2),ว(76),า(1)	0	-	1	-	2	ฐ(2)
3	ฐ(7)	4	ฉ(1),ฮ(24)	5	ฐ(20)	6	-
7	ฐ(1)	8	ฐ(1)	9	-	ๅ	1(33)
โ	-	ไ	โ(3),ไ(90)	ใ	-	ๅ	โ(4)
ๆ	ฦ(10),ศ(1)	๐	ล(1),ค(1)	๑	ต(1)	๒	-
๓	ศ(8)	๔	-	๕	ฮ(1)	๖	ใ(1)
๗	ฮ(1),๒(1)	๘	8*	๙	ฐ(2),ฮ(1)		

10.1) แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มตัวอักษร (*ดูตารางที่ 10.2)

ตัวอักษร	ผลลัพธ์*	ตัวอักษร	ผลลัพธ์*
ก	ค(1),ก(1),ถ(3),ภ(3)	ค	ค(38),ล(34),ต(8),ศ(1)
ศ	พ(1),ส(25),พ(13)	ห	ท(7),พ(61),ฬ(24)
๘	ฮ(1),๒(3),๗(4)		

10.2) แสดงตัวอย่างผลลัพธ์ที่นำไปทดสอบการแยกตัวอักษรจากตารางที่ 10.1

แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มตัวอักษรด้วยกระดาษถ่ายเอกสาร ที่ความสว่าง 50 เปอร์เซ็นต์ของโครงข่าย A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดสอบโครงข่าย A พบว่าค่าเปอร์เซ็นต์ความถูกต้องของบางกลุ่มมีค่าน้อย ทั้งนี้เนื่องจากค่าถ่วงน้ำหนักที่ได้จากการสอนมีค่าซึ่งไม่เหมาะสม เช่น

ตัวอักษร ข กระดาษพิมพ์เลเซอร์ที่ความสว่าง 50 % จำนวนตัวอักษรผิด 81 ตัว

ตัวอักษร ฏ กระดาษพิมพ์เลเซอร์ที่ความสว่าง 50 % จำนวนตัวอักษรผิด 59 ตัว

ตัวอักษร ใ กระดาษถ่ายเอกสารที่ความสว่าง 100 % จำนวนตัวอักษรผิด 29 ตัว

ตัวอักษร ร กระดาษถ่ายเอกสารที่ความสว่าง 50 % จำนวนตัวอักษรผิด 44 ตัว

ตัวอักษร เ กระดาษถ่ายเอกสารที่ความสว่าง 50 % จำนวนตัวอักษรผิด 33 ตัว

จากนั้นทำการปรับค่าเทรสโฮลด์ของตัวอักษรบางกลุ่มแล้วทำการสอนโครงข่ายใหม่ ซึ่งขอเรียกว่า โครงข่าย B ตารางที่ 11 แสดงค่าเทรสโฮลด์ที่ใช้สอนโครงข่าย B

ตารางที่ 11

ตัวอักษร	ค่า Threshold	ตัวอักษร	ค่า Threshold	ตัวอักษร	ค่า Threshold	ตัวอักษร	ค่า Threshold
ก	45	ฃ	60	ฆ	35	ค	60
ฅ	65	ฉ	30	ง	10	จ	30
ฉ	10	ช	45	ซ	45	ฌ	55
ญ	20	ฎ	20	ฏ	35	ฐ	35
ฑ	25	ฒ	35	ณ	45	ด	60
ต	55	ถ	40	ท	45	ธ	55
น	10	บ	10	ป	10	ผ	10
ฝ	10	พ	55	ฟ	10	ภ	40
ม	35	ย	35	ร	50	ฤ	50
ล	40	ฦ	40	ว	35	ศ	55
ษ	25	ส	20	ห	50	ฬ	55
อ	45	ฮ	35	ฯ	10	ะ	5
า	35	0	30	1	15	2	2
3	10	4	10	5	2	6	10
7	10	8	10	9	10	เ	25
โ	35	ไ	70	ใ	65	ๅ	15
ๆ	10	๐	10	๑	30	๒	10
๓	10	๔	10	๕	10	๖	10
๗	10	๘	10	๙	10		

แสดงค่าเทรสโฮลด์ที่ใช้ในการสอนโครงข่าย B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 12 แสดงผลการสอนของโครงข่าย B ซึ่งจำนวนโหนดของชั้นคอมเพททิทิพที่ได้จากการสอนมีจำนวน 166 โหนด โดยโครงข่ายสามารถจดจำโครงข่ายได้ในครั้งที่ 12 ตารางที่ 13 แสดงจำนวนโหนดที่มีในแต่ละกลุ่มตัวอักษร

ตารางที่ 12

ครั้ง	ค่า α	จำนวนโหนด	จำนวนตัวอย่างที่ถูกต้อง	% ความถูกต้องของการสอนตัวอย่าง
0	-	75	-	-
1	0.8	145	1476	98.40
2	0.8	157	1486	99.06
3	0.8	162	1490	99.33
4	0.6	164	1494	99.59
5	0.6	165	1496	99.73
6	0.6	166	1496	99.73
7	0.6	166	1492	99.46
8	0.4	166	1497	99.80
9	0.4	166	1499	99.93
10	0.4	166	1498	99.86
11	0.4	166	1498	99.86
12	0.2	166	1500	100
13	0.2	166	1500	100
14	0.2	166	1500	100
15	0.2	166	1500	100

เปอร์เซ็นต์การเรียนทุก ๆ 4 รอบใช้จำนวนข้อมูลที่นำมาสอน 1500 ตัว ของโครงข่าย B

ตารางที่ 14 แสดงผลการทดสอบการวิเคราะห์ระหว่างโครงข่าย B กับข้อมูลที่ใช้ในการทดสอบโครงข่าย A ซึ่งจากตารางแสดงให้เห็นว่าค่าของเปอร์เซ็นต์ของโครงข่าย B มีเปอร์เซ็นต์สูงกว่า เนื่องจากทิศทางของค่าถ่วงน้ำหนักที่เพิ่มขึ้นในชั้นคอมเพททิทิพของโครงข่าย B ที่ได้หลังการปรับค่าเทรสโฮลด์จากโครงข่าย A มีทิศทางคลอบคลุมพื้นที่ของตัวอย่างมากกว่า ตารางที่ 15 ถึง 18 แสดงจำนวนตัวอักษรที่แยกแยะผิดของแต่ละกลุ่มของโครงข่าย B ซึ่งจากตารางนี้แสดงให้เห็นว่ามีจำนวนตัวอักษรที่ตกผิดน้อยกว่าเดิม เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอักษร ข กระดาษพิมพ์เลเซอร์ที่ความสว่าง 50 % มีจำนวนตัวผิด 2 ตัว

ตัวอักษร ฎ กระดาษพิมพ์เลเซอร์ที่ความสว่าง 50 % มีจำนวนตัวผิด 0 ตัว

ตัวอักษร ใ กระดาษถ่ายเอกสารที่ความสว่าง 100 % มีจำนวนตัวผิด 5 ตัว

ตัวอักษร ร กระดาษถ่ายเอกสารที่ความสว่าง 50 % มีจำนวนตัวผิด 19 ตัว

ตัวอักษร เ กระดาษถ่ายเอกสารที่ความสว่าง 50 % มีจำนวนตัวผิด 1 ตัว

ตารางที่ 13

ตัวอักษร	จำนวนโหนด	ตัวอักษร	จำนวนโหนด	ตัวอักษร	จำนวนโหนด	ตัวอักษร	จำนวนโหนด
ก	3	ข	4	ฃ	1	ค	5
ฅ	5	ฆ	2	ง	1	จ	2
ฉ	1	ช	3	ซ	3	ฌ	4
ญ	2	ฎ	2	ฏ	2	ฐ	1
ฑ	1	ฒ	3	ณ	4	ด	4
ต	6	ถ	3	ท	2	ธ	3
น	1	บ	1	ป	1	ผ	1
ฝ	1	พ	2	ฟ	1	ภ	2
ม	1	ย	2	ร	4	ฤ	4
ล	2	ฦ	3	ว	2	ศ	2
ษ	2	ส	2	ห	3	ฬ	2
อ	4	ฮ	1	ฯ	1	ะ	1
า	2	0	2	1	4	2	2
3	2	4	1	5	2	6	2
7	3	8	1	9	2	ิ	2
ี	3	ึ	3	ื	5	ุ	2
ู	2	ฺ	1	฿	2	เ	1
ๅ	1	็	1	่	1	้	2
๊	1	๋	2	์	1		

แสดงจำนวนโหนดในชั้นคอมเพททิทิพของแต่ละกลุ่มตัวอักษรของโครงข่าย B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 14

ชนิดกระดาษที่ทดสอบ (%ความสว่าง)	จำนวนตัวอักษรที่ผิด	เปอร์เซ็นต์ความถูก
พิมพ์ laser (100 %)	21	99.86
พิมพ์ laser (50 %)	398	97.34
ถ่ายเอกสาร (100%)	48	99.68
ถ่ายเอกสาร (50%)	646	95.69

แสดงผลการทดสอบการแยกตัวอักษรภาษาไทยชุดละ 15,000 ตัว โดยใช้โครงข่าย B

ตารางที่ 15

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

แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษพิมพ์เลเซอร์
ที่ความสว่าง 100 เปอร์เซ็นต์ของโครงข่าย B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 16

ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด
ก	17*	ช	ช(2)	ซ	ฌ(1),พ(1)	ค	57*
ต	42*	ม	16*	ง	-	จ	ฉ(7),ช(3)
ฉ	-	ร	ช(3)	ช	-	ฌ	-
ญ	-	ฎ	ฎ(1)	ฎ	-	ฐ	-
ท	ฒ(5)	ฒ	-	ณ	ฌ(38)	ด	ต(30)
ต	-	ถ	-	ท	พ(41)	ธ	-
น	บ(1)	บ	-	ป	-	ผ	-
ฝ	-	พ	ท(51),ห(2)	ฟ	-	ภ	ฦ(4)
ม	บ(1)	ย	อ(1)	ร	ฐ(17)	ฤ	-
ล	-	ฦ	-	ว	-	ศ	-
ษ	-	ส	-	ห	พ(16),ฟ(19)	ฬ	-
อ	-	ฮ	ต(1)	ฯ	-	ะ	-
๑	ว(1)	๐	-	๑	-	๒	-
๓	-	๔	ฮ(9)	๕	-	๖	-
๗	-	๘	-	๙	-	๑	-
โ	ใ(4)	ไ	ไ(9)	ใ	-	ๅ	-
ๆ	-	๐	-	๑	-	๒	-
๓	-	๔	-	๕	-	๖	-
๗	-	๘	๗(1)	๘	ศ(3)		

16.1) แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มตัวอักษร (*ดูตารางที่ 16.2)

ตัวอักษร	ผลลัพธ์*	ตัวอักษร	ผลลัพธ์*
ก	ค(2),ถ(8),ท(1),ภ(4),ศ(2)	ค	ท(15),ค(14),ต(27),ศ(1)
ต	ค(2),ต(40)	ม	ฒ(11),ณ(4),พ(1)

16.2) แสดงตัวอย่างผลลัพธ์ที่นำไปทดสอบการแยกตัวอักษรจากตารางที่ 16.1

แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มตัวอักษรด้วยกระดาษพิมพ์เลเซอร์ที่ความสว่าง 50 เปอร์เซ็นต์ของโครงข่าย B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากทิศทางของค่าถ่วงน้ำหนักที่เพิ่มขึ้นของโครงข่าย B นี้ครอบคลุมพื้นที่ของกลุ่มตัวอย่างที่นำมาทดสอบดีกว่าโครงข่าย A แต่ทิศทางของค่าถ่วงน้ำหนักที่เพิ่มขึ้นนี้อาจไปใกล้ทิศทางของกลุ่มอื่นมากเกินจึงทำให้เกิดการแยกแยะผิด เช่น โหนด อ ของโครงข่าย B ที่เพิ่มจากโครงข่าย A จากจำนวน 3 โหนดเป็น 4 โหนด ในการทดสอบการแยกแยะตัวอักษร ย ด้วยกระดาษถ่ายเอกสารที่ความสว่าง 50 เปอร์เซ็นต์ โครงข่าย A มีตัวอักษร อ ตกจำนวน 1 ตัว แต่ในโครงข่าย B มีตัวอักษร อ ตกจำนวน 5 ตัว

ตารางที่ 17

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

แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษถ่ายเอกสาร
ที่ความสว่าง 100 เปอร์เซ็นต์ของโครงข่าย B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 18

ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด
ก	8*	ช	ช(5)	ซ	ซ(1)	ค	78*
ค	ค(8),ค(1),ต(30)	ฌ	ฒ(1),ฌ(3)	ง	-	จ	ฉ(2),ช(1)
ฉ	ข(1),อ(1)	ซ	ซ(9)	ซ	ฒ(6),พ(21)	ฌ	ฒ(1)
ญ	-	ฎ	ฎ(1)	ฎ	ฎ(2)	ฐ	-
ฑ	ฒ(8)	ฒ	-	ณ	ณ(51)	ค	ค(41)
ต		ถ	อ(4)	ท	พ(53)	ธ	ข(2)
น	ฑ(3),บ(3)	บ	-	ป	-	ผ	-
ฝ	ฟ(1)	พ	-	ฟ	ป(1)	ภ	ฦ(5)
ม	บ(8),ย(1)	ย	ร(1),อ(5)	ร	ช(19)	ฤ	ถ(3)
ล	-	ฦ	ภ(1)	ว	ช(1)	ศ	พ(1),ฬ(3)
ษ	พ(6)	ส	ศ(1)	ห	65*	ฬ	พ(3)
อ	-	ฮ	ต(3)	ฯ	ศ(1)	ะ	ช(1)
า	ว(27)	0	-	1	-	2	ฐ(2)
3	ฐ(7)	4	ฉ(1),ฮ(24)	5	ฐ(20)	6	ศ(1)
7	ฐ(1)	8	ฐ(1)	9	-	่	1(1)
์	-	ๆ	ใ(48)	ไ	-	ๅ	า(1),โ(4)
ๆ	ต(2),ฦ(3),ศ(1)	๐	ล(1),๑(1)	๑	ต(1)	๒	-
๓	ฒ(13)	๔	-	๕	ช(1)	๖	ไ(1)
๗	ช(1),๒(1)	๘	ช(1),๒(3),๗(4)	๙	ฐ(1),ช(1),ศ(3)		

18.1) แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มตัวอักษร (*ดูตารางที่ 18.2)

ตัวอักษร	ผลลัพธ์*	ตัวอักษร	ผลลัพธ์*
ก	ค(1),ค(1),ถ(3),ฦ(3)	ค	ค(26),ค(25),ต(20),ศ(1)
ห	ท(6),พ(32),ฬ(27)		

18.2) แสดงตัวอย่างผลลัพธ์ที่นำไปทดสอบการแยกตัวอักษรจากตารางที่ 18.1

แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษถ่ายเอกสาร
ที่ความสว่าง 50 เปอร์เซ็นต์ของโครงข่าย B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการเพิ่มค่าเทรสโฮลด์ขึ้นบางค่าของโครงข่าย B ส่งผลให้โครงข่ายสามารถแยกแยะตัวอักษรได้มากขึ้น ดังนั้นในโครงข่าย C ที่สร้างใหม่นี้จะทำการปรับค่าเทรสโฮลด์ของโครงข่าย B เพื่อให้มีการเพิ่มโหนดมากขึ้น และลดค่าเทรสโฮลด์บางตัวอักษร เช่น ข เพื่อให้ผลของการเพิ่มโหนดมีจำนวนไม่มาก ตารางที่ 19 แสดงค่าเทรสโฮลด์ที่กำหนดให้กับโครงข่าย C (ซึ่งเพิ่มจากค่าเทรสโฮลด์ของโครงข่าย B บางค่า)

ตารางที่ 19

ตัวอักษร	ค่า Threshold	ตัวอักษร	ค่า Threshold	ตัวอักษร	ค่า Threshold	ตัวอักษร	ค่า Threshold
ก	45	ช	55	ข	35	ค	60
ฆ	65	ฆ	30	ง	10	จ	30
ฉ	10	ช	45	ซ	45	ฌ	55
ญ	20	ฎ	20	ฏ	35	ฐ	35
ฑ	25	ฒ	35	ณ	45	ด	60
ต	55	ถ	40	ท	45	ธ	60
น	10	บ	10	ป	10	ผ	10
ฝ	10	พ	60	ฟ	10	ภ	40
ม	35	ย	35	ร	50	ฤ	50
ล	40	ฦ	40	ว	35	ศ	60
ษ	25	ส	20	ห	55	พ	65
อ	45	ฮ	35	ฯ	10	ะ	5
า	35	0	30	1	15	2	2
3	10	4	10	5	2	6	10
7	10	8	10	9	10	ๅ	25
โ	35	ไ	70	ใ	65	ๆ	15
ๆ	10	๐	10	๑	30	๒	10
๓	๋	๔	10	๕	10	๖	10
๗	10	๘	10	๙	10		

แสดงค่าเทรสโฮลด์ที่ใช้ในการสอนโครงข่าย C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 20

ครั้ง	ค่า α	จำนวนโหนด	จำนวนตัวอย่างที่ถูกต้อง	% ความถูกต้องของการสอนตัวอย่าง
0	-	75	-	-
1	0.8	150	1476	98.4
2	0.8	165	1488	99.19
3	0.8	171	1490	99.33
4	0.6	173	1494	99.59
5	0.6	174	1497	99.80
6	0.6	174	1490	99.33
7	0.6	174	1494	99.59
8	0.4	174	1497	99.80
9	0.4	174	1499	99.93
10	0.4	174	1498	99.86
11	0.4	174	1500	99.86
12	0.2	174	1500	100.00
13	0.2	174	1500	100.00
14	0.2	174	1500	100.00
15	0.2	174	1500	100.00

เปอร์เซ็นต์การเรียนทุก ๆ 4 รอบ ใช้จำนวนข้อมูล
ที่นำมาสอน 1500 ตัวของโครงข่าย C

ตารางที่ 20 แสดงเปอร์เซ็นต์ความถูกต้องในการเรียนรู้ของ CPN ซึ่งมีจำนวนโหนดมากกว่าโครงข่าย A และ B คือ 174 โหนด ตารางที่ 21 แสดงจำนวนโหนดที่ของแต่ละกลุ่ม โดยมีโหนดของค่าถ่วงน้ำหนักบางกลุ่มมีจำนวนน้อยลงเช่น ข เนื่องจากมีการกำหนดค่าเทรสโฮลด์น้อย ตารางที่ 23-26 แสดงผลการแยกแยะตัวอักษรตามกลุ่มต่างๆ

ตารางที่ 22 แสดงผลการทดสอบการแยกแยะที่นำข้อมูลจากการทดสอบข้างต้น ซึ่งจากผลการทดสอบค่าเปอร์เซ็นต์ความถูกต้องมีค่าลดลงเมื่อเทียบกับผลการแยกแยะของโครงข่าย B การทดลองกระดาษถ่ายเอกสารที่ความสว่าง 100 เปอร์เซ็นต์ (ตารางที่ 25) ของโครงข่าย C มีผลการแยกแยะเหมือนโครงข่าย B เนื่องจากทิศทางของค่าถ่วงน้ำหนักของโหนดที่เพิ่มขึ้นในโครงข่าย C อยู่ในทิศทางเดียวกับโครงข่าย B ที่สามารถแยกแยะข้อมูลได้แล้ว จึงเป็นผลให้ความสามารถของโหนดที่เพิ่มขึ้นของโครงข่าย C มีเปอร์เซ็นต์การแยกแยะเท่ากับโครงข่าย B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 21

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

แสดงจำนวนโหนดในชั้นคอมเพททิทิพของแต่ละกลุ่มตัวอักษรของโครงข่าย C

ตารางที่ 22

ชนิดกระดาษที่ทดสอบ (%ความสว่าง)	จำนวนตัวอักษรที่ผิด	เปอร์เซ็นต์ความถูก
พิมพ์ laser (100 %)	21	99.86
พิมพ์ laser (50 %)	405	97.29
ถ่ายเอกสาร (100%)	48	99.68
ถ่ายเอกสาร (50%)	664	95.57

แสดงผลการทดสอบการแยกตัวอักษรภาษาไทยชุดละ 15000 ตัว ของโครงข่าย C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 28

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

แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษพิมพ์เลเซอร์
ที่ความสว่าง 100 เปอร์เซ็นต์ของโครงข่าย C

การทดลองกระดาษถ่ายเอกสารที่ความสว่าง 50 เปอร์เซ็นต์ (ตารางที่ 26) ของโครงข่าย C มีผลการแยกแยะน้อยกว่าโครงข่าย B เนื่องจากทิศทางของค่าถ่วงน้ำหนักที่เพิ่มขึ้นนี้ไปเข้าใกล้ทิศทางของกลุ่มอื่นมากเกินจึงทำให้เกิดการแยกแยะผิด เช่น โหนด พ ของโครงข่าย C ที่เพิ่มจากโครงข่าย B จากจำนวน 2 โหนดเป็น 4 โหนด (ตารางที่ 13 และ 21) ผลการทดสอบโครงข่าย C มีตัวอักษร ท ตกจำนวน 65 ตัวและโครงข่าย B มีตัวอักษร ท ตกจำนวน 53 ตัว เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 24

ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด
ก	17*	ช	ช(2)	ซ	ซ(1),ฒ(1)	ค	57*
ค	42*	ฌ	ฒ(11),ฌ(4)	ง	-	จ	ฉ(7),ช(3)
ฉ	-	ช	ช(3)	ซ	-	ฌ	-
ญ	-	ฎ	ฎ(1)	ฏ	-	ฐ	-
ฑ	ฒ(5)	ฒ	-	ณ	ณ(38)	ค	ศ(30)
ต	-	ถ	-	ท	ท(46)	ธ	-
น	บ(1)	บ	-	ป	-	ผ	-
ฝ	-	พ	ท(52),ห(1)	ฟ	-	ภ	ภ(4)
ม	บ(1)	ย	อ(1)	ร	ธ(22)	ฤ	-
ล	-	ฦ	-	ว	-	ศ	ห(1)
ษ	-	ส	-	ห	ห(14),ห(16)	ห	-
อ	-	ฮ	ต(1)	ฯ	-	ะ	-
า	ว(1)	0	-	1	-	2	-
3	-	4	อ(3)	5	-	6	-
7	-	8	-	9	-	ๆ	-
โ	ใ(4)	ไ	ใ(9)	ใ	-	ไ	-
ๆ	-	๐	-	๑	-	๒	-
๓	-	๔	-	๕	-	๖	-
๗	-	๘	๗(1)	๘	ศ(3)		

24.1) แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มตัวอักษร (*ดูตารางที่ 24.2)

ตัวอักษร	ผลลัพธ์*	ตัวอักษร	ผลลัพธ์*
ก	ค(2),ศ(1),ถ(8),ท(1),ภ(4),ศ(1)	ค	ค(15),ค(14),ต(27),ศ(1)
ค	ค(2),ต(40)		

24.2) แสดงตัวอย่างผลลัพธ์ที่นำไปทดสอบการแยกตัวอักษรจากตารางที่ 24.1

แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มตัวอักษรด้วยกระดาษพิมพ์เลเซอร์ที่ความสว่าง 50 เปอร์เซ็นต์ของโครงข่าย C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากที่กล่าวข้างต้นการสอนโครงข่าย C มีการลดค่าเทรสโฮลด์บางค่าจากโครงข่าย B ซึ่งได้แก่ กลุ่มของ ข ซึ่งส่งผลให้จำนวนโหนดของกลุ่ม ข ในโครงข่าย C ลดลงจากเดิมคือ จาก 4 โหนดเหลือ 3 โหนด แต่เมื่อนำมาทดสอบแยกแยะกับกระดาษแบบต่างๆแล้วสามารถแยกแยะตัวอักษรได้จำนวนเท่ากัน เนื่องจากทิศทางของค่าถ่วงน้ำหนักของกลุ่ม ข ของโครงข่าย C ยังครอบคลุมพื้นที่ของตัวอย่างที่นำมาสอนเช่นเดียวกับโครงข่าย B แม้ว่าจำนวนโหนดจะมีจำนวนน้อยกว่า

ตารางที่ 25

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

แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษถ่ายเอกสาร
ที่ความสว่าง 100 เปอร์เซ็นต์ของโครงข่าย C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 26

ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด	ตัวอักษร	จำนวนการแยกแยะผิด
ก	8*	ช	ช(5)	ฃ	ช(1)	ค	78*
ค	ค(8),ค(1),ต(30)	ฌ	ฒ(1),ณ(3)	ง	-	จ	ฉ(2),ช(1)
ฉ	ข(1),อ(1)	ซ	ซ(9)	ฌ	ฒ(6),พ(19)	ญ	ฒ(1)
ญ	-	ฎ	ฏ(1)	ฏ	ฎ(2)	ฐ	-
ฑ	ฒ(8)	ฒ	-	ณ	ฌ(51)	ด	ต(41)
ต	-	ถ	อ(4)	ท	พ(65)	ธ	ข(2)
น	ฑ(3),บ(3)	บ	-	ป	-	ผ	-
ฝ	ฟ(1)	พ	-	ฟ	ป(1)	ภ	ฦ(5)
ม	บ(8),ย(1)	ย	ร(1),อ(5)	ร	ช(24)	ฤ	ถ(3)
ฉ	-	ฦ	ภ(1)	ว	ช(1)	ศ	ฟ(7)
ษ	พ(6)	ส	ศ(1),ฟ(3)	ห	65*	พ	พ(3)
อ	-	ฮ	ต(3)	ฯ	ศ(1)	ะ	ช(1)
า	ว(27)	0	-	1	-	2	ฐ(2)
3	ฐ(7)	4	ฉ(1),ฮ(24)	5	ฐ(20)	6	-
7	ฐ(1)	8	ฐ(1)	9	-	เ	1(1)
โ	-	ใ	ไ(48)	ไ	-	ๆ	า(1),โ(4)
ๆ	ฦ(3),ศ(5)	๐	ถ(1),๑(1)	๑	ต(1)	๒	-
๓	ฒ(13)	๔	-	๕	ช(1)	๖	ไ(1)
๗	ช(1),๒(1)	๘	ช(1),๒(3),๗(4)	๙	ฐ(1),ช(1),ศ(3)		

26.1) แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มตัวอักษร (*ดูตารางที่ 26.2)

ตัวอักษร	ผลลัพธ์*	ตัวอักษร	ผลลัพธ์*
ก	ค(1),ค(1),ถ(3),ฦ(3)	ค	ค(26),ค(25),ต(20),ศ(1)
ห	ท(6),พ(32),ฟ(27)		

26.2) แสดงตัวอย่างผลลัพธ์ที่นำไปทดสอบการแยกตัวอักษรจากตารางที่ 26.1

แสดงจำนวนตัวอักษรที่แยกแยะผิดในแต่ละกลุ่มด้วยกระดาษถ่ายเอกสาร
ที่ความสว่าง 50 เปอร์เซ็นต์ของโครงข่าย C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองของโครงข่าย A,B และ C สามารถสรุปให้เห็นถึงผลการเพิ่มโหนดของชั้นคอมเพททิทิพได้ว่า โหนดที่เพิ่มขึ้นนี้สามารถช่วยให้โครงข่ายสามารถแยกแยะตัวอักษรได้ดีขึ้น กรณีการเพิ่มโหนดแล้วทำให้ขอบเขตของพื้นที่ของโหนดผู้ชนะที่มีขนาดกว้างมากมีขนาดแคบลง แต่มีหลายพื้นที่ ซึ่งจากกรณีของโครงข่าย A และ B จำนวนโหนดของค่าถ่วงน้ำหนักที่เพิ่มของโครงข่าย B เนื่องจากค่าถ่วงน้ำหนักบางโหนดของโครงข่าย A มีขอบเขตของพื้นที่โหนดผู้ชนะกว้างมาก ดังนั้นเมื่อมีการเพิ่มโหนดในโครงข่าย B จึงช่วยทำให้ขอบเขตของพื้นที่โหนดผู้ชนะมีขนาดแคบลงแต่มีจำนวนโหนดมากขึ้นเป็นผลให้การแยกแยะตัวอักษรของโครงข่าย B ดี แต่การเพิ่มโหนดของค่าถ่วงน้ำหนักอาจทำให้ผลของการแยกแยะน้อยลงได้ กรณีที่โหนดที่เพิ่มขึ้นนั้นมีทิศทางเดียวกับโหนดที่สามารถแยกแยะข้อมูลได้ดีอยู่แล้ว (อยู่ในทิศทางที่เหมาะสม) และโหนดที่เพิ่มขึ้นมีทิศทางใกล้กับกลุ่มข้อมูลอื่นมากเกิน จึงเป็นผลให้เกิดการแยกแยะผิด เช่น กรณีของโครงข่าย B และ C ซึ่งจำนวนโหนดของโครงข่าย C มีจำนวนมากกว่ากลับให้ผลการแยกแยะที่น้อยกว่า เนื่องจากจำนวนโหนดของค่าถ่วงน้ำหนักที่เพิ่มของโครงข่าย C มีทิศทางเดียวกับโครงข่าย B ที่สามารถแยกแยะข้อมูลได้ดีแล้ว อีกทั้งโหนดของค่าถ่วงน้ำหนักที่เพิ่มขึ้นมีทิศทางใกล้กับกลุ่มข้อมูลอื่นๆมาก

<u>การทดสอบการแยกแยะตัวอักษรที่อยู่ในรูปของประโยค</u>

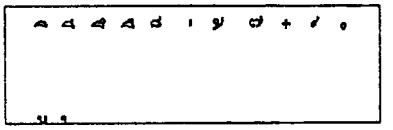
การทดลองนี้มีจุดประสงค์เพื่อทดสอบการแยกแยะตัวอักษรภาษาไทยที่อยู่ในรูปของประโยค ซึ่งการทดลองนี้ได้แบ่งโครงข่ายออกเป็น 3 โครงข่าย คือ

1. โครงข่ายที่ใช้แยกแยะตัวอักษรภาษาไทยระดับบน (สระบน และ วรรณยุก)
2. โครงข่ายที่ใช้แยกแยะตัวอักษรภาษาไทยระดับกลางบรรทัด
3. โครงข่ายที่ใช้แยกแยะตัวอักษรภาษาไทยระดับล่าง (สระล่าง)

โครงข่ายที่ใช้แยกแยะตัวอักษรภาษาไทยระดับกลางบรรทัดนำค่าถ่วงน้ำหนักจากการทดลอง 5.1 โดยเลือกจากโครงข่าย B เนื่องจากให้ค่าผลลัพธ์จากทดสอบดีที่สุด ส่วนโครงข่ายที่ใช้แยกแยะตัวอักษรภาษาไทยบนและล่างนี้ได้ทำการสอนโครงข่ายเพิ่ม เมตริกที่เป็นข้อมูลเข้าของโครงข่ายทั้งสองมีขนาด 24x24 ภาพที่ 41 แสดงตัวอักษรของที่ใช้สอนโครงข่ายบนและล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 41



แสดงตัวอักษรของที่ใช้สอนโครงข่ายบนและล่าง

การทดลองนี้ทำบนเครื่อง 486-DX40 โดยข้อมูลที่นำมาทดสอบนี้นำจากกระดาษที่พิมพ์ผ่านเครื่องเลเซอร์ ซึ่งในการทดสอบจะมีการหมุนเอกสารให้เอียงเป็นมุมต่างๆ คือ 1,2,3 องศาเพื่อทดสอบประสิทธิภาพการวิเคราะห์ตัวอักษร เนื่องจากขั้นตอนของการใช้งานจริงนั้นผู้ใช้อาจไม่สังเกตเห็นว่าภาพหน้าเอกสารที่สแกนนั้นหมุน

ขั้นตอนของกาทดสอบนี้ประกอบด้วยขั้นตอน 1.การแยกภาพตัวอักษรออกจากภาพประโยค,2.การคัดลอกตัวอักษรออกจากประโยค,3.การแยกแยะตัวอักษร,4.การจดเรียงตัวอักษร ซึ่งขั้นตอนที่ 1,2 และ 4 ในวิทยานิพนธ์นี้จะไม่ขอกล่าวถึงเนื่องจากไม่ได้เป็นเนื้อหาที่มุ่งเน้น (ภาคผนวก ก,ข และ ค ตามลำดับ) ภาพที่ 42 แสดงตัวอย่างเอกสารที่นำมาใช้ในการทดสอบ และตารางที่ 27 แสดงผลการแยกแยะตัวอักษรโดยการทดลองนี้ใช้ข้อมูลทดสอบ 994 ตัวอักษร

ตารางที่ 27

องศาการหมุน	เปอร์เซ็นต์ความถูกต้อง	
	ซ้าย	ขวา
0	88.6	88.6
1	84.8	86.8
2	78.6	80.2
3	67.2	77.4

แสดงผลการทดสอบแยกแยะตัวอักษรในประโยค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการทดลองผลของความผิดพลาดเนื่องจากการเซกเมนต์ภาพตัวอักษรผิดเฉลี่ยประมาณ 11 เปอร์เซ็นต์ และเวลาที่ใช้ในการแยกแยะตัวอักษรประมาณ 6 ตัวอักษรต่อวินาที

ภาพที่ 42

เมื่อเวลา 06.45 น.วันที่ 7 สิงหาคมมีเหตุวางระเบิดขึ้นที่ จ.นราธิวาส โดย ร.ต.ประสิทธิ์ ชูโชติ ผู้บำนวยการศูนย์พลศึกษาและกีฬาจังหวัดนราธิวาส เข้าแจ้งความที่ สภ.อ.เมืองนราธิวาสว่า จะมีที่ดินตรวจตราพื้นที่โดยรอบอาคารสำนักงานศึกษาธิการจังหวัดนราธิวาส ซึ่งอยู่ในรั้วเดียวกับศาลากลาง จ.นราธิวาส พบว่าที่ด้านหลังอาคารสำนักงานศึกษาธิการจังหวัดใกล้เครื่องทำความเย็นมีระเบิดชนิดแสวงเครื่องวางอยู่ จึงให้ไปตรวจสอบและทำการกู้ระเบิดด้วย

หลังจากได้รับรายงานแล้ว นายสวัสดิ์ กฤตรัชตนันท์ ผู้ว่าราชการ จ.นราธิวาส พล.ต.ต.ธรรมนูญ ทับเสถียร ผบก.ภน.ตร.ภ.จว.นราธิวาส พ.ต.อ.กมล โพธิชาย ผกก.สภ.อ.เมืองนราธิวาส พร้อมกำลังตำรวจจำนวนหนึ่งรุดไปที่เกิดเหตุพบกระป๋องนมผงขนาดบรรจุ 1 กิโลกรัมภายในมีดินระเบิดและเชื้อประทุไฟฟ้า เชื่อมต่อกับนาฬิกาและถ่านไฟฉาย 3 ก้อนเป็นตัวจุดเชื้อประทุ นอกจากนี้ ยังพบสีสเปรย์อีก 1 กระป๋อง

ภายหลังกู้ระเบิดเสร็จแล้วพบว่า ระเบิดแสวงเครื่องดังกล่าวอยู่ในถุงกระดาษถูกตั้งเวลาระเบิดไว้ที่เวลา 05.00 น. แต่เจ้านาฬิกาหยุดเดินเมื่อเวลา 02.50 น. เจ้าหน้าที่สำรวจได้ผ่ากระป๋องนมที่บรรจุดินระเบิดภายในประกอบด้วยดินระเบิดชนิด ซีโฟร์หนัก 1 ใน 4 ปอนด์ปั้นในเตรผสมน้ำมันไขสาร์น้ำหนัก 1.4 กิโลกรัม สะปุชนาด 3 นิ้วจำนวน 128 ตัว จึงเก็บไว้เป็นหลักฐาน คาดว่าหากระเบิดทำงานจะทำให้อาคารสำ

แสดงตัวอย่างภาพของรูปประโยคที่นำมาใช้ในการทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

บทสรุปและแนวทางการพัฒนาในอนาคต

บทสรุป

การนำ CPN มาใช้ในการแยกแยะตัวอักษรนี้ใช้วิธีการสอนตัวอย่างข้อมูลภาพของตัวอักษรให้กับโครงข่าย ลักษณะการเรียนรู้ของ CPN จะพยายามสร้างค่าถ่วงน้ำหนักเฉลี่ยให้กับตัวอย่างที่นำมาสอน ซึ่งลักษณะของค่าถ่วงน้ำหนักที่ได้เปรียบเสมือนตาราง look-up table ที่ใช้ในการเปรียบเทียบกับข้อมูลที่นำเข้ามาสอนโครงข่าย

แม้ว่างานวิจัยนี้ได้มีการใช้วิธีการเพิ่มโหนดของชั้นคอมเพททิทิพในขณะสอนโครงข่ายเพื่อแก้ปัญหาการสอนโครงข่ายให้สามารถแยกแยะตัวอย่างที่นำมาสอนได้นั้น แต่วิธีที่ใช้นี้เป็นวิธีการกำหนดค่าเทรสโฮลด์ซึ่งเป็นเปอร์เซ็นต์ความเหมือนระหว่างเวคเตอร์ของค่าถ่วงน้ำหนักกับอินพุตเวคเตอร์ที่เข้าสู่โครงข่าย ซึ่งการกำหนดนี้ต้องมีการพิจารณาเนื่องจากถ้ามีกำหนดค่าให้สูง (ถ้าเวคเตอร์ของค่าถ่วงน้ำหนัก กับ อินพุตเวคเตอร์ เหมือนกันน้อยกว่าค่าเทรสโฮลด์ให้ทำการเพิ่มโหนด) จำนวนโหนดของค่าถ่วงน้ำหนักที่เพิ่มในชั้นคอมเพททิทิพก็จะเพิ่มมาก ซึ่งการกำหนดให้ค่าเทรสโฮลด์มีค่ามากควรให้กับกลุ่มของตัวอักษรที่มีลักษณะคล้ายๆกัน เช่น ค,ต,ค,ต,ข,ช,ฌ,ฌ เพราะขอบเขตของกลุ่มข้อมูลมีระยะที่ใกล้ จากการทดลองบทที่ 5 เรื่องการแยกแยะตัวอักษรทีละชุดของตัวอักษร แสดงเห็นว่าการโครงข่ายมีจำนวนโหนดมากบางครั้งก็ไม่สามารถทำให้ผลของการแยกแยะดีขึ้น เนื่องจากถ้าค่าเวคเตอร์ของค่าถ่วงน้ำหนักของโหนดที่เพิ่มขึ้นมีทิศทางเดียวกับค่าเวคเตอร์ของค่าถ่วงน้ำหนักที่มีทิศทางที่เหมาะสมอยู่แล้ว และทิศทางของค่าถ่วงน้ำหนักของโหนดที่เพิ่มขึ้นมีระยะใกล้กับข้อมูลของกลุ่มอื่นเกินกว่าทิศทางของค่าถ่วงน้ำหนักของกลุ่มข้อมูลนั้นๆ

ด้วยเทคนิคหลักที่ใช้ในการสอนโครงข่ายคือ การเพิ่มโหนดในชั้นคอมเพททิทิพด้วยวิธีการกำหนดค่าเปอร์เซ็นต์ของความเหมือนระหว่างค่าถ่วงน้ำหนักกับอินพุตเวคเตอร์ที่เข้าสู่โครงข่าย และการแยกสอนทีละกลุ่มข้อมูลช่วยให้โครงข่ายสามารถเรียนรู้ตัวอย่างได้รวดเร็ว กล่าวคือการเพิ่มจำนวนโหนดทำให้ขอบเขตของพื้นที่ของโหนดผู้ชนะของแต่ละโหนดมีขนาดแคบลงซึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลดปัญหาการเรียนผิดโหนดกรณีที่ขอบเขตของข้อมูลใกล้เคียงกันและแก้ปัญหาการแกว่งของทิศทางของค่าถ่วงน้ำหนัก เนื่องจากขอบเขตของข้อมูลมีขนาดกว้างมากดังนั้นเมื่อมีตัวอย่างข้อมูลที่มีระยะห่างกันมากเข้ามาสอนโครงข่ายทำให้ทิศทางของค่าถ่วงน้ำหนักมีการแกว่งมากเป็นผลการปรับค่าถ่วงน้ำหนักสู่สมดุลย์ได้ยาก ส่วนการแยกสอนนอกจากลดปัญหาการเรียนผิดโหนดแล้ว ยังช่วยลดเวลาของการสอนเนื่องจากการหาโหนดผู้ชนะการคำนวณจะคิดเพียงแค่ในกลุ่มของข้อมูลที่สอนอยู่เท่านั้นจึงช่วยลดเวลาในการคำนวณ ซึ่งจากการทดลองการสอนโครงข่าย A,B และ C ในบทที่ 5 เรื่องการทดลองการแยกแยะตัวอักษรทีละชุดตัวอักษรใช้เวลาเฉลี่ยประมาณ 35 นาที โดยจำนวนรอบที่ใช้สอนโครงข่าย 60 รอบ

การนำเทคนิคการเพิ่มจำนวนโหนดของชั้นคอมเพททิทิพนอกจากแก้ปัญหาการเรียนผิดโหนดแล้ว ยังช่วยให้โครงข่ายสามารถเรียนรู้ตัวอย่างใหม่ๆได้อย่างไม่จำกัดเนื่องจากการเพิ่มโหนดของ CPN เป็นแบบไดนามิค ซึ่งเมื่อเทียบกับการเรียนรู้ด้วยวิธีที่กำหนดจำนวนโหนดไว้ตายตัวแล้ว อาจจดจำตัวอย่างที่สอนใหม่ไม่ได้เนื่องจากโครงข่ายมีจำนวนโหนดของค่าถ่วงน้ำหนักที่ใช้ในการจดจำไม่เพียงพอ

ประสิทธิภาพการแยกแยะตัวพิมพ์อักษรภาษาไทยของโครงข่าย CPN เมื่อนำมาทดสอบการแยกแยะกับกระดาษแบบต่างๆมีผลที่น่าพอใจ จากการทดลองในบทที่ 5 หัวข้อการทดลองการแยกแยะตัวอักษรทีละชุดตัวอักษรโครงข่ายที่ให้ผลดีที่สุดสามารถแยกแยะได้ถูกต้องประมาณ 98.1 เปอร์เซ็นต์ (โครงข่าย B) และโครงข่ายที่ให้ผลน้อยที่สุด (โครงข่าย A) ประมาณ 96.95 เปอร์เซนต์ซึ่งเป็นเปอร์เซ็นต์ที่สูง ภาพที่ 43 แสดงตัวอย่างภาพของตัวอักษรที่โครงข่ายสามารถแยกแยะได้แสดงให้เห็นว่าประสิทธิภาพของโครงข่าย CPN มีค่อนข้างสูงมากเนื่องจากคุณภาพของภาพที่นำมาทดสอบบางภาพมีคุณภาพที่ค่อนข้างต่ำ ซึ่งจากรูปภาพตัวอักษรบางภาพใช้มนุษย์ในการแยกแยะยังค่อนข้างยาก ส่วนภาพที่ 44 แสดงตัวอย่างภาพตัวอักษรที่โครงข่าย CPN ไม่สามารถแยกแยะได้

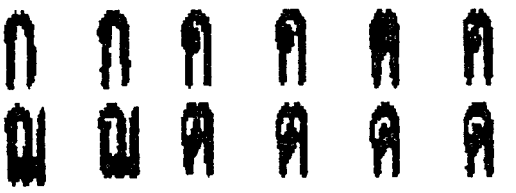
ภาพที่ 43

43.1) พิมพ์ laser สว่าง 100 % ก,ก,ก,ก,ก,ก,ฎ,ฎ,ค,ค,ห,ห

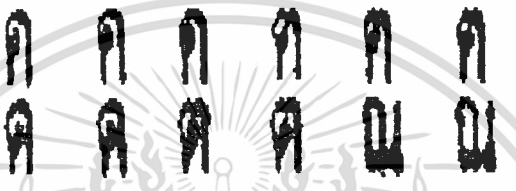
แสดงตัวอย่างภาพตัวอักษรที่โครงข่าย CPN สามารถแยกแยะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 43 (ต่อ)



43.2) พิมพ์ laser สว่าง 50 % ก,ก,ก,ก,ก,ก,ฌ,ฌ,ค,ค,ค,ค



43.3) ถ่ายเอกสาร สว่าง 100 % ค,ค,ค,ค,ค,ค,ค,ค,ต,ต,ฌ,ฌ

43.4) ถ่ายเอกสาร สว่าง 50 % ค,ค,ค,ค,ฌ,ฌ,ฎ,ฎ,ฎ,ฎ,ค,ค

แสดงตัวอย่างภาพตัวอักษรที่โครงข่าย CPN สามารถแยกแยะได้

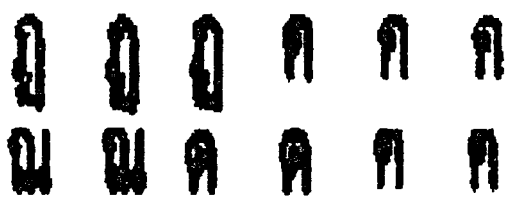
ภาพที่ 44

44.1) พิมพ์ laser สว่าง 100 % ฎ(ฎ),ฎ(ฎ),ค(ค),ค(ค),ห(ห),(ค),ค(ค),
ค(ค),ค(ค),ค(ค)

แสดงตัวอย่างภาพตัวอักษรที่โครงข่าย CPN ไม่สามารถแยกแยะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 44 (ต่อ)



44.2) พิมพ์ laser สว่าง 50 % ฏ(ฏ),ฏ(ฏ),ฏ(ฏ),ค(ค),ค(ค),ค(ค),ฒ(ฒ),
ฒ(ฒ),ค(ค),ค(ค),ค(ค),ค(ค)

44.3) ถ่ายเอกสาร สว่าง 100 % ช(ข),ข(ช),ข(ช),ต(ค),ค(ต),ค(ค),ค(ค),
ค(ค),ฏ(ฏ),ฏ(ฏ)

44.4) ถ่ายเอกสาร สว่าง 50 % ค(ต),ค(ค),ค(ค),ค(ต),ข(ช),ข(ช),(ฒ),
ฒ(ฒ),ค(ต),ค(ต),ฏ(ฏ),ฏ(ฏ)

แสดงตัวอย่างภาพตัวอักษรที่โครงข่าย CPN ไม่สามารถแยกแยะได้

จากการทดสอบเวลาที่ใช้ในการแยกแยะตัวอักษรด้วยเครื่อง 486-DX2 ใช้เวลาประมาณ 6 ตัวอักษรต่อวินาที ซึ่งถือว่าเป็นเวลาที่น้อยมากเมื่อเทียบกับการพิมพ์ดีดของมนุษย์

จากที่กล่าวข้างต้นเนื่องจากการเรียนของ CPN เหมือนการสร้างตาราง look-up table ดังนั้นเมื่อมีการนำข้อมูลเข้าสู่โครงข่ายเพื่อทำการวิเคราะห์ในตำแหน่งที่เลื่อนจากตำแหน่งที่สอนโครงข่ายหรือมีขนาดใหญ่กว่าข้อมูลที่นำมาสอนมากอาจทำให้โครงข่ายไม่สามารถแยกแยะข้อมูลนั้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แนวทางการพัฒนาในอนาคต

งานวิจัยนี้มุ่งเน้นเรื่องการแยกแยะตัวอักษรซึ่งเป็นส่วนหนึ่งของระบบเครื่องอ่านตัวอักษรด้วยแสง จากการทดลองการวิเคราะห์หรือแยกแยะตัวอักษรที่อยู่ในรูปประโยคในบทที่ 5 พบว่ามีความผิดพลาดที่เกิดจากการเซกเมนต์ตัวอักษรผิดเนื่องจากภาพของตัวอักษรติดกัน ซึ่งปัญหานี้ถือได้ว่าเป็นปัญหาที่สำคัญในการพัฒนาเครื่องอ่านตัวอักษรด้วยแสง เพราะก่อนที่จะนำภาพตัวอักษรไปใช้แยกแยะตัวอักษรต้องผ่านกระบวนการเซกเมนต์ภาพตัวอักษรก่อน ซึ่งถ้าภาพตัวอักษรที่ผ่านการเซกเมนต์ผิดจะเป็นผลให้การวิเคราะห์หรือแยกแยะตัวอักษรผิดด้วย ดังนั้นแนวทางการพัฒนาระบบเครื่องอ่านตัวอักษรด้วยแสงควรมีการศึกษาวิธีการเซกเมนต์ภาพตัวอักษรที่มีความสามารถสูง เพื่อลดปัญหาที่เกิดจากการเซกเมนต์ภาพตัวอักษรผิด

รูปแบบตัวอักษรที่ใช้ในงานวิจัยนี้เป็นแบบฟอนท์เดียว (single font) แต่ในปัจจุบันรูปแบบตัวอักษรในหน้าเอกสารมีหลายแบบ ดังนั้นถ้าต้องการนำ CPN ไปใช้ในการวิเคราะห์หรือแยกแยะตัวอักษรแบบหลายฟอนท์ (multi font) ควรทำการหาลักษณะรูปร่าง (feature) ของตัวอักษรก่อนเพื่อนำลักษณะรูปร่างของตัวอักษรมาสอนโครงข่าย ซึ่งวิธีนี้ทำให้โครงข่ายมีขนาดเล็กและประหยัดหน่วยความจำกว่าการที่นำภาพของตัวอักษรหลายแบบมาสอนในโครงข่ายเดียว เพราะทำให้เวลาที่ใช้ในการคำนวณนาน เนื่องจากจำนวนโหนดของค่าถ่วงน้ำหนักมีมากและยังใช้หน่วยความจำมากด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. R:G. Casey and C.R. Jim, "A processor-Based OCR System", IBM Journal of research and development. no.27 (1983) pp. 386-341.
2. James A. Freeman and David M. Skapura, Neural Networks Algorithms, Applications, and Programming Techinques. Massachusetts : Addison-Wesley, 1991.
3. Laurene Fausett. Fundamentals of Neural Networks Architectures, Algorithms, and Applications. New Jersey : Prentice Hall International Inc, 1994.
4. G.F.Page, J.B. Gomm and D.Williams. Applicaton of neural networks to moddelling and control. England : Chapman & Hall, 1993.
5. Philip D.Wasserman. Advanced Methdos in Neural Computing. New York : Van Nostrand Reinhold, 1993.
6. Maureen Caudill and Charles Butler. Understanding neural networks:Computer Explorations. Massachusetts : The MIT press, 1993
7. Robert Hecht-Nielsen. Neurocomputing. Massachusetts : Addison-Wesley, 1990.
8. Igor Aleksander and Helen Morton. An Introduction to Neural Computing, London : Chapman & Hall, 1992.
9. Valluru B. Rao and Hayagriva V. Rao. C++ Neural Networks and Fuzzy Logic. USA : MIS press, 1993.
10. ชม กิ้มปาน. การรู้จำรูปแบบอักษรพิมพ์ภาษาไทยของคอมพิวเตอร์. วิทยานิพนธ์ วิศวกรรมศาสตร์ดุษฎีบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2529.
11. ประสาร ตังตสานนท์. การจดจำรูปแบบตัวอักษรคัดลายมือภาษาไทย โดยใช้วิธีแยกลักษณะเด่น. วิทยานิพนธ์ วิศวกรรมศาสตร์มหาบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2529.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

12. สุรพันธ์ เอื้อไพบูลย์. การจดจำอักษรลายมือเขียนภาษาไทยโดยการพิจารณาหัวตัวอักษร. วิทยานิพนธ์ วิศวกรรมศาสตร์มหาบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบัน-เทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2531.

13. ชาย เกษมอมรกุล. การออกแบบพจนานุกรมสำหรับการเรียนรู้อักษรลายมือไทย-อังกฤษอัตโนมัติบนเครื่องไมโครคอมพิเตอร์. วิศวกรรมศาสตร์มหาบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2532.

14. สุรสิทธิ์ ราตรี. การรู้จำอักษรตัวพิมพ์ภาษาไทยโดยวิธีค้นหาลักษณะโครงสร้างลายเส้น. วิทยานิพนธ์วิศวกรรมศาสตร์มหาบัณฑิต. กรุงเทพฯ : บัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 2532.

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

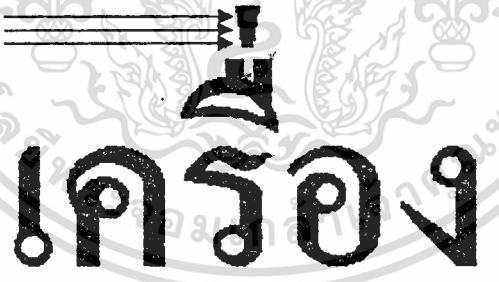
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

การแยกภาพตัวอักษรออกจากภาพประโยค

การแยกภาพตัวอักษรออกจากภาพประโยค ขั้นแรกทำการตรวจกวาดจุดภาพในแนวนอนจากมุมบนซ้ายมือไปขวามือ เมื่อพบจุดดำ (มีค่าเป็น 1) ก็จะทำการดึงภาพตัวอักษรด้วยวิธีติดตามรอยขอบ (ภาคผนวก ข) เพื่อหาขอบภาพของตัวอักษรแล้วเก็บขนาดความสูงของภาพอักษรไว้ ภาพที่ 45 แสดงการตรวจกวาดตามแนวนอน

ภาพที่ 45



แสดงการตรวจกวาดจุดดำของภาพตามแนวนอนในการแยกภาพตัวอักษร

จากหลักการวิธีการวิเคราะห์หน้าเอกสารของ คุณสมศักดิ์ วลัยรัชต์[15] ด้วยวิธีทำให้เบลอ ทำให้ข้อมูลภาพที่อยู่ใกล้กันชิดกันเป็นกลุ่มภาพเดียวกัน ภาพที่ 46 แสดงตัวอย่างการทำเบลอซึ่งทำให้ภาพตัวอักษรที่อยู่ใกล้ชิดกันเป็นกลุ่มข้อมูลภาพชุดเดียวกัน จากเหตุผลนี้หลังการทำในขั้นตอนแรก ให้ทำการตรวจกวาดข้อมูลภาพในแนวตั้งโดยเริ่มจากขอบล่างซ้ายของภาพประมาณ 3-4 แถว จนถึงขอบล่างขวาของภาพ ซึ่งเมื่อพบจุดดำแล้วทำการดึงภาพตัวอักษรที่พบใหม่ด้วยวิธีติด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตามรอยขอบเพื่อหาขอบภาพของตัวอักษรแล้วเก็บขนาดความสูงของภาพอักษรดังเช่นขั้นตอนแรก ย้อนกลับมาทำขั้นตอนนี้ใหม่จนกระทั้งการตรวจกวาดในแนวตั้งไม่พบจุดสีดำ ซึ่งวิธีนี้ทำให้ได้ข้อมูลภาพตัวอักษรตามประเภทของตัวอักษร คือ สระบนกับวรรณยุก (ิ , ี , ็ , ่ , ้), ตัวอักษรระดับกลางบรรทัด (ก,ข,ป,ฏ,โ,ใ) และ สระล่าง (ุ , ู) โดยภาพตัวอักษรที่มีความสูงมากสุดเป็นตัวอักษรระดับกลางบรรทัด,ภาพที่มีตำแหน่งน้อยกว่าตำแหน่งของภาพตัวอักษรกลางบรรทัดเป็นสระบนกับวรรณยุก และภาพที่มีตำแหน่งมากกว่าตำแหน่งภาพของตัวอักษรกลางบรรทัดเป็นสระล่าง

เมื่อทราบตำแหน่งของภาพตัวอักษรระดับกลางบรรทัดให้เก็บตำแหน่งกลางบรรทัดของภาพตัวอักษร เพื่อนำไปเปรียบเทียบกับระยะการตรวจกวาดในแนวนอนว่ามีระยะมากกว่าหรือไม่ โดยตั้งสมมุติฐานว่าถ้าการตรวจกวาดแนวนอนมีระยะมากกว่าตำแหน่งกลางบรรทัดของภาพอักษรกลาง ถือว่าการแยกภาพจากบรรทัดนั้นสมบูรณ์

ภาพที่ 46

46.1) แสดงเอกสารก่อนทำเบลอ 46.2) แสดงเอกสารหลังทำเบลอ

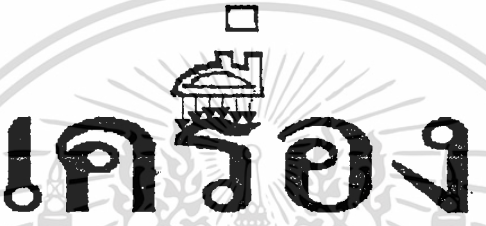
แสดงผลของการตัดบรรทัดด้วยวิธีการทำเบลอ

ภาพที่ 47.1 และ 47.2 แสดงการตรวจกวาดตามแนวตั้งของคำว่า “รี่” ภาพที่ 47.1 แสดงการตรวจกวาด (ี) ตามแนวตั้งหลังจากการตรวจกวาดในแนวนอน (่) และภาพที่ 47.2 แสดงการตรวจกวาด (ร) ตามแนวตั้งหลังจากการตรวจกวาด (ี) ภาพที่ 47.3 แสดงการตรวจสอบ “เ” ตามแนวนอนซึ่งเป็นขั้นตอนแรก เนื่องจากเมื่อมีการตรวจกวาดตามแนวตั้งของ “ร” แล้วไม่พบจุดสีดำเลยจึงย้อนกลับทำขั้นตอนแรกใหม่ ภาพที่ 48 แสดงลำดับรูปตัวอักษรที่แยกด้วยวิธีนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 47

47.1) แสดงกวาดตรวจกวาดแนวตั้งของภาพ (ี) หลังตรวจกวาด (')

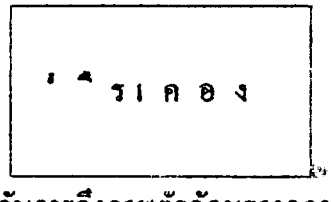


47.2) แสดงกวาดตรวจกวาดแนวตั้งของภาพ (ร) หลังตรวจกวาด (ี)

47.3) แสดงกวาดตรวจกวาดแนวนอนของภาพ (เ) เนื่องจาก
ไม่มีภาพในแนวตั้งของ (ร)

แสดงการแยกภาพคำว่า "รี" จากภาพประโยค

ภาพที่ 48



แสดงผลลำดับการดึงภาพตัวอักษรจากภาพประโยค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

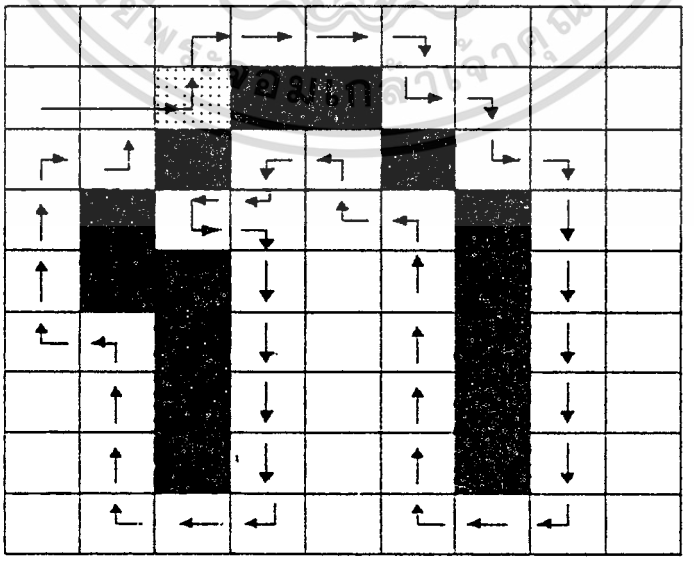
ภาคผนวก ข

เทคนิคการติดตามรอยขอบของภาพ (Contour Following)

เทคนิคการติดตามรอบขอบภาพ[15] ในวิทยานิพนธ์ฉบับนี้ได้ถูกนำมาใช้ในการคัดลอกส่วนของภาพตัวอักษร การทำงานของเทคนิคนี้เป็นการเดินไต่ไล่ไปตามขอบระหว่างส่วนที่เป็นรูปภาพกับส่วนที่เป็นพื้นเบื้องหลัง โดยการตรวจกวาดจะเริ่มจากจุดมุมบนซ้ายของข้อมูลภาพ ตรวจกวาดไปในทิศทางจากซ้ายไปขวาและเลื่อนลงจากบนลงล่าง การตรวจกวาดเมื่อพบจุดใดๆของภาพที่เป็นจุดคำ (มีค่าเป็น 1) ก็จะเปลี่ยนลักษณะการเคลื่อนที่ไปยังจุดภาพจุดถัดไปเสียใหม่

ภาพที่ 49 แสดงการทำงานของเทคนิคการติดตามรอยขอบภาพ ซึ่งแสดงการเคลื่อนไปตามจุดต่างๆของขอบภาพ โดยที่ตำแหน่งสีเทาเป็นจุดเริ่มต้น

ภาพที่ 49



แสดงลักษณะการติดตามรอยขอบของภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

การจัดเรียงรูปประโยคหลังการวิเคราะห์ [15]

ตัวอักษรแต่ละตัวที่ถูกแยกออกจากภาพประโยค ก่อนที่จะส่งผ่านไปทำการแยกแยะตัวอักษรจะถูกบันทึกค่าตำแหน่งพิกัด เนื่องจากการคัดลอกภาพตัวอักษรออกจากภาพประโยคทำให้ทราบว่าลำดับบรรทัดของแต่ละประโยคแล้วจึงไม่มีการจัดเรียงลำดับตำแหน่งตามค่า y_{min} ของภาพตัวอักษร เพื่อเป็นการจัดเรียงตัวอักษรให้อยู่ในรูปประโยคจึงทำการจัดเรียงลำดับตัวอักษรตามค่า x_{min} จากค่าน้อยไปมาก ตัวอย่างการจัดเรียงตัวอักษรในประโยค

ขั้นตอนที่ 1 ตัวอย่างข้อความเดิมก่อนถูกแยกตัวอักษรออกประโยค

คอมพิวเตอร์กราฟิคเป็นเทค

ขั้นตอนที่ 2 เมื่อผ่านขั้นตอนการแยกตัวอักษรออกจะแยกตัวอักษรในแต่ละบรรทัดออกมาโดยมีลำดับดังนี้

์ป ์ร ิพ ิฟคอมวเตอกราคเนเทค

ขั้นตอนที่ 3 จากขั้นตอนที่ 2 เมื่อมีการเรียงตำแหน่งที่ถูกแยกตามลำดับ x_{min} จะได้ตัวอักษร “ค” เป็นตัวอักษรตัวแรกแล้วตามด้วยตัวอักษร “อ” เรียงต่อกันไปจนหมดบรรทัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง
บทความทางวิชาการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรู้จำตัวพิมพ์อักษรไทยโดยใช้ Counterpropagation Neural Network (CPN)

Printed Thai Characters Recognition using Counterpropagation Neural Network (CPN)

ดร. บุญธีร์ เครือตราชู * นาย อภิรักษ์ จิรายุลกุล**

บทคัดย่อ

งานวิจัยนี้นำ CPN มาประยุกต์ในงานจดจำตัวอักษรภาษาไทย โดยเปลี่ยนแปลงโครงสร้างหลัก 2 ส่วน ส่วนแรกคือวิธีการเรียนรู้โดยใช้วิธีการแยกสอนตัวอย่าง input ทีละกลุ่ม ส่วนที่สองคือโครงสร้างใน competitive layer ของ CPN สามารถเพิ่มจำนวนโหนดในขั้นตอนการเรียนรู้ ซึ่งช่วยลดเวลาการเรียนรู้และความผิดพลาดในการจดจำตัวอักษร งานวิจัยนี้จะแสดงให้เห็นถึงปัญหาและวิธีการแก้ไขโครงสร้างของ CPN เพื่อช่วยในการจดจำตัวอักษรภาษาไทย ผลการทดสอบการจดจำตัวอักษรภาษาไทย 75 ตัว ซึ่งใช้ข้อมูลอย่างละ 15,000 ตัวที่พิมพ์จากเครื่อง laser และจากกระดาษถ่ายเอกสารซึ่งได้ผลความถูกต้องประมาณ 99 % โดยข้อมูลตัวอย่างที่นำมาสอนจำนวน 1500 ตัว

Abstract

A modified neural network is presented. It is used in a Thai character recognizer. The network is modified from CPN in two main path. The first part is in the learning method where group separation learning is used. The second path,which is the main modified part,is in the network architecture where the number of nodes in the CPN competitive layer can be increased during the learning phase. This will help reduce the error in character recognition. The problem of CPN in application to Thai character recognition and the modification are presented. Then,the new modified network is used to test 75 Thai characters recognition from 15,000 laser printed samples and 15,000 copied printed samples with the recognition rate up to 99 % ,using 1,500 training characters.

* อาจารย์ที่ปรึกษา ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

** นักศึกษาปริญญาโท ภาควิชาคณิตศาสตร์ คณะวิทยาศาสตร์ สาขาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

1.บทนำ

ในการสร้างระบบจดจำอักษรภาษาไทยกระบวนการที่จำเป็นกระบวนการหนึ่งก็คือ การวิเคราะห์ตัวอักษร งานวิจัยนี้ได้นำวิธีการของนิวรัลเน็ตเวิร์ก (neural network) มาใช้ในการวิเคราะห์ตัวอักษรภาษาไทย นิวรัลเน็ตเวิร์กมีคุณสมบัติที่เรียกว่า *generalization* ซึ่งคุณสมบัตินี้ทำให้นิวรัลเน็ตเวิร์กสามารถวิเคราะห์ input ที่ไม่เคยเห็นได้โดยใช้วิธีการเรียนรู้ตัวอย่าง input เท่านั้น ตัวแบบที่ใช้ในงานวิจัยนี้คือ forward counter-propagation network (CPN) ซึ่งมีโครงสร้างแบบ multilayer คือ input,competitive และ output layer การเรียนรู้ของ CPN เป็นแบบ unsupervised learning เวลาที่ใช้สอนโครงข่ายชนิดนี้เมื่อเทียบกับ Backpropagation network ซึ่งมีลักษณะการเรียนรู้เป็นแบบ supervised learning แล้วใช้เวลาเร็วกว่ามากและมีผลความถูกต้องใกล้เคียงกัน จากการทดลองใช้เวลาประมาณ 30 นาทีจากตัวอย่างที่นำมาสอน 1500 ตัวโครงข่ายสามารถจดจำตัวอย่างได้หมด การเรียนรู้ของ CPN จะปรับตัวในลักษณะการสร้างค่า weight เฉลี่ยของตัวอย่างของ input ที่นำมาใช้สอน

2.สถาปัตยกรรมของ CPN ที่ใช้ในงานวิจัย

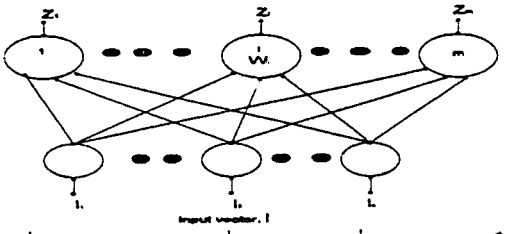
สถาปัตยกรรมของ CPN [1] ที่นำมาใช้ในงานวิจัยนี้ประกอบด้วย 2 ส่วนคือ input layer และ competitive layer ดังแสดงในรูปที่ 1 Input Layer มีจำนวน n โหนด (node) แต่ละโหนดรับข้อมูลจาก input vector $\mathbf{I} = (I_1, I_2, \ldots, I_n)$ แล้วทำการกระจาย component ของ $\mathbf{I}$ ให้กับ Competitive Layer ซึ่งมีจำนวน m โหนด output vector ของ layer นี้คือ $\mathbf{Z}=(z_1, z_2, \ldots, z_m)$ ซึ่ง z_i มีค่าเท่ากับ 1 ถ้าค่า net input ของโหนดที่ i มีค่ามากสุดซึ่งเรียก **โหนดผู้ชนะ** (winner node) และค่า output ของโหนดที่น้อยกว่า winner node มีค่าเท่ากับ 0 สมการที่ 1 แสดงการหาค่า net input

$$net_i = \sum_j^n I_j w_{ij} \quad \ldots\ldots(1)$$

โดยที่ I_j เป็นค่า element ตัวที่ j ของ input

w_{ij} เป็นค่า weight ของ j ตัวโหนดที่ I

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1 สถาปัตยกรรมของ CPN ที่ใช้ในงานวิจัย โดยที่ $I = (I1,I2,....In)$ เป็น input vector ที่เข้าสู่ input layer , $W = (W1,W2,....,Wm)$ เป็นโหนด (node) ของ weight และ $Z = (Z1,Z2,...Zm)$ เป็น output vector ที่ออกจาก competitive layer

3.วิธีการเรียนรู้ของ CPN (CPN learning)

งานวิจัยนี้จะกล่าวถึงการเรียนรู้ของ competitive layer เพียงอย่างเดียว การเรียนรู้ของ layer นี้จะเลือกปรับ weight ของ winner node เพียงโหนดเดียว โดยใช้สมการที่ 2 [2]

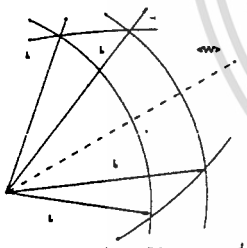
$$w(t+1) = \alpha\left(I - w(t)\right) \quad \text{.....(2)}$$

โดยที่ I : เป็น input vector

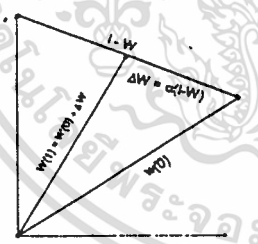
α : learning rate ซึ่งมีค่าน้อยกว่า 1

w : weight ที่เวลา t

การสอนโครงข่ายนี้จะทำจนกระทั่งได้ weight ที่เป็นค่าเฉลี่ยของตัวอย่าง input ที่นำมาสอนโครงข่าย รูปที่ 2 แสดงลักษณะของ weight ที่ปรับจนได้ค่าเฉลี่ย รูปที่ 3 แสดงทิศทางการปรับ weight



รูป 2 แสดงการเรียนจนได้ weight เฉลี่ย



รูป 3 แสดงทิศทางการปรับของ weight

4. Algorithm ที่ใช้สอน CPN [3]

Step 0. initialize weights,learning rates (α)

Step 1. While stopping condition is False,do steps 2-7

Step 2. for each training input I, do step

Step 3. Set I input activation to vector *I*.

Step 4. Find winning cluster unit

(using equation 1) call index *J*

Step 5. update weights for unit *Zj*

(using equation 2)

Step 6. Reduce learning rate (α)

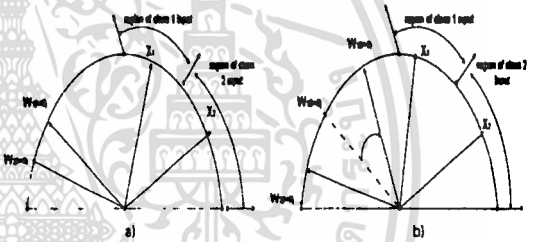
Step 7. if weight is average or learning rate <= 0

stopping condition is True

5. ปัญหาการใช้ CPN แยกตัวอักษรภาษาไทย

5.1 ปัญหาการเรียนผิดโหนด (stuck weight)

การเรียนของ CPN จะเลือกปรับ weight ของโหนดที่มีค่า net input มากที่สุดเพียงโหนดเดียว ดังนั้นอาจเกิดปัญหาการเรียนผิดโหนดซึ่งเรียกว่า stuck weight [2] กรณีแรกเนื่องจากตัวอักษรภาษาไทยมีรูปร่างคล้ายกัน เช่น ก,ถ,ภ,ค,ด,ต,ค ซึ่งเมื่อผ่านการ scan จะยิ่งทำให้ภาพของตัวอักษรมีความคล้ายกันมาก เพราะฉะนั้น weight ของตัวอักษรจึงมีระยะใกล้กันมาก และกรณีที่สองเกิดจากการกำหนดค่าของ weight เริ่มต้นไม่ดี รูปที่ 4 แสดงลักษณะของ stuck weight กรณีกำหนดค่า weight เริ่มต้นไม่ดี รูป 4 (a) จะเห็นว่า W1 อยู่ใกล้กับ input class X1 และ X2 มากกว่า W2 ดังนั้นเมื่อมี input ของ class X2 เข้ามาทำให้ W1 ถูกปรับดังแสดงในรูปที่ 4 (b)

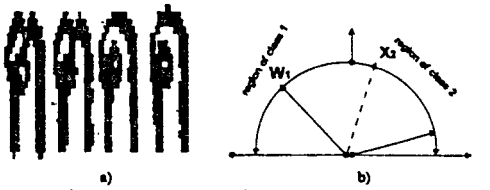


รูปที่ 4 a) แสดงการกำหนดค่าเริ่มต้นของ weight ที่ทำให้เกิด stuck weight

b) แสดง weight หลังการปรับ weight รูปที่ 4 แสดงการเกิด stuck weight

5.2 ปัญหาการหาค่า weight เฉลี่ย

ภาพของตัวอักษรที่ได้จากการ scan นั้นมีความหลากหลายและรูปร่างของตัวอักษรภาษาไทยมีความคล้ายกันมาก รูปที่ 5 (a) แสดงภาพตัวอักษร ค,ต แบบต่าง ๆ ที่ได้จากการ scan ซึ่งแสดงให้เห็นว่าขอบเขตของ win region ของแต่ละ class ของตัวอักษรมีกว้างมากและมีขอบเขตใกล้เคียงกัน ดังนั้นการสอน CPN เพื่อให้ได้ค่า weight เฉลี่ยจึงเป็นสิ่งที่ยากมาก เพราะขึ้นอยู่กับจำนวนครั้งในการสอนโครงข่ายหรือจำนวนตัวอย่างที่นำมาใช้ในการสอน รูปที่ 5 (b) แสดงปัญหาการหาค่าเฉลี่ยของ weight ผิดของ W2 ซึ่งเมื่อมี input X2 เข้ามาจะไปตกที่ W1 ซึ่งผิดเพราะอยู่ใกล้ input X2 มากกว่า [2]



รูปที่ 5 (a) แสดงตัวอย่างภาพที่ได้จาก scan

(b) แสดงปัญหาการหาค่าเฉลี่ยของweight ผิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.การสอน CPN เพื่อการจดจำอักษรไทย

6.1 การกำหนดค่าเริ่มต้นของ weight

การกำหนดค่าเริ่มต้นของ weight ด้วยวิธีการสุ่ม (random) อาจทำให้เกิดปัญหาในข้อ 5.1 ได้ เพราะตัวอักษรภาษาไทยมีลักษณะคล้ายกัน เช่น เมื่อสอน input ก แล้ว weight จะปรับตัวเองให้มีลักษณะเหมือน ก จากนั้นนำ input ค มาสอน โหนดผู้ชนะอาจตกที่โหนด input ก ได้เพราะในโครงข่ายอาจไม่มีโหนดใดมีลักษณะใกล้เคียงกับ ค เลยเมื่อเปรียบเทียบกับโหนดที่ input ก ไปตก งานวิจัยนี้ได้กำหนดลักษณะค่า weight ดังนี้

6.1.1 กำหนดค่าเริ่มต้นของ weight เป็นค่าของ input vector ที่ถูก normalized

จากปัญหาการกำหนดค่า weight เริ่มต้น งานวิจัยนี้ใช้วิธีการนำ input ที่นำมาสอนมาเป็นต้นแบบของ weight แต่เนื่องจากตัวอักษรภาษาไทยมีลักษณะคล้ายกัน การให้ค่าเริ่มต้นของ weight มีค่าเท่ากับ input ที่อยู่ในรูป 1-0 อาจทำให้เกิดปัญหาการแยกตัวอักษรเช่น กรณีของ ถ หรือ ฤ จะสังเกตเห็นว่าส่วนหัวของตัวอักษรเหมือนกัน ดังนั้นเมื่อมี input ถ เข้ามาค่า net input ของ weight ฤ อาจเป็นผู้ชนะได้ ดังนั้นการกำหนดค่าเริ่มต้นของ weight จึงใช้ค่าของ input ที่ผ่านการ normalized ดังสมการที่ 3 [2]

$$\theta i = I_i \left(\sum_i I_i \right)^{-1} \quad .....(3)$$

กำหนดให้ I_i เป็นค่าของ element ตัวที่ i สำหรับ input จำนวน n ตัว และเรียก vector ของ $(\theta_1, \theta_2, \ldots, \theta_n)^t$ ว่า **reflectance pattern** ซึ่ง $\sum \theta_i = 1$ การทำ normalized นี้ทำให้ค่าของ element ของ input vector มีความสัมพันธ์กับ input ที่เข้ามา

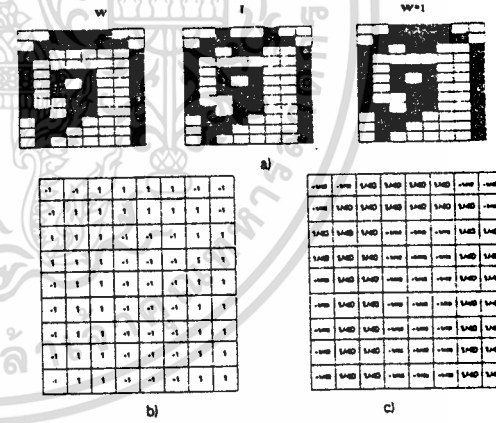
6.1.2 กำหนดให้พื้นของ weight มีค่าติดลบ

เนื่องจากโครงสร้างของ CPN มีเพียง layer เดียว ดังนั้นจึงไม่มีส่วนใดของโครงข่ายที่ช่วยยับยั้งหรือบอกว่า input ที่เข้ามาในโหนด (node) ต่าง ๆ นั้นไม่ใช่ input ของ region ของ weight นั้น ปัญหาที่เห็นได้ชัดเจนที่สุดก็คือ การแยกตัวอักษรระหว่าง ถ,ฤ ถึงแม้ว่าจะใช้วิธี normalized ในการกำหนดค่า weight เริ่มต้นแล้ว แต่ยังไม่สามารถแก้ปัญหาได้เพราะว่าปริมาณเนื้อตัวอักษรของ ถ มีน้อยกว่า ฤ จึงทำให้ weight ของ ถ ค่ามากกว่า ดังนั้นเมื่อมี input vector ฤ เข้ามาอาจทำให้ weight ของ ถ เป็นผู้ชนะได้ กรณีนี้การให้พื้นของ weight มีค่าติดลบจึงช่วยยับยั้งส่วนที่เกินของ ฤ ใน weight ถ ได้

การกำหนดค่าติดลบให้กับ weight นี้นับเป็นสิ่งที่ยุ่งยากมาก ถ้ากำหนดให้ค่าติดลบมีค่าน้อยเกินก็อาจทำให้ส่วนที่ติดลบไม่สามารถยับยั้งส่วนที่มีค่าเป็นเนื้อตัวอักษรได้ งานวิจัยนี้จึง ***กำหนดให้ค่าติดลบมีค่าเท่ากับค่า normalized ของ input*** รูป 6 (c) แสดงเมตริกของ weight เริ่มต้นที่ใช้ใน CPN

6.2 การใช้ค่า input ลักษณะ +1,-1

การกำหนด weight ให้มีค่าบวก-ลบช่วยแก้ปัญหากรณีที่ขนาดของเนื้อ input มีขนาดใหญ่กว่า weight ดังเช่นกรณีหัวข้อ 6.1.2 แต่กรณีที่ขนาดเนื้อของ input vector ของตัวอักษรมีน้อยกว่า weight ไม่ได้ถูกนำมาคิด ดังนั้นการกำหนดให้ input ที่เข้าสู่โครงข่ายมีค่า +1,-1 จึงช่วยแก้ปัญหากรณีได้ เนื่องจากค่าลบของ input จะไปยับยั้งส่วนของตัวอักษรที่มีเนื้อขาด ปัญหาที่เห็นได้ชัดคือ การแยกตัวอักษร ด,ต เพราะลักษณะตัวอักษรมีความแตกต่างกันที่ตรงส่วนหยักบนส่วนหัวของตัวอักษร ซึ่งภาพที่ได้จากการ scan บางครั้งส่วนหยักของตัวอักษรอาจมีน้อยมาก ดังนั้นการคิดค่า net-input จากส่วนที่เกินเพียงอย่างเดียวอาจไม่เพียงพอ



รูปที่ 6 a) แสดงการ map ของ weight ด กับ input ด, b) แสดงเมตริกของ input vector ของ ก c) เมตริกของ weight เริ่มต้นของ ก

รูปที่ 6 (a) แสดงตัวอย่าง input vector ด (I) และ weight ของ ด (W) ซึ่งลักษณะที่คล้ายกันมาก ดังนั้นเมื่อนำ input vector ที่มีค่า +1,-1 ของ ด มา map กับ weight ของ ด จะทำให้ส่วนที่ขาดของ input ด ยับยั้งส่วนที่เกินของ weight ด ไม่ให้ active รูปที่ 6 (b) แสดงเมตริกของ input vector ที่เข้าสู่โครงข่าย

6.3 การคิดค่า net input

เนื่องจากพื้นของ input vector และ weight vector มีค่าลบ เพราะฉะนั้นการหาค่า net-input จึงต้องไม่นำค่าที่เป็นลบของพื้น input และ weight มาคิดด้วยกัน ดังนั้นจึงต้องมีการตรวจสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
วิธีการตรวจสอบ
IF Wi > 0
   net-input = Ii*Wi
ELSE
  IF I > 0
     net-input = Ii * Wi
    ENDIF
ENDIF
```

เนื่องจากหลังการปรับ weight ทำให้ผลรวมของ weight มีค่าไม่เท่ากับหนึ่ง ดังนั้นการคิดค่า net-input จึงต้องนำค่าผลรวมของ weight ที่มีค่าเป็นบวกมาหาร เพื่อให้ผลรวมของ weight มีค่าเป็นหนึ่ง ซึ่งเขียนแทนด้วยสมการที่ 4

$$net_i = \left(\sum_j I_j w_{ij}\right) / \sum_j w1_j \quad \text{.....(4)}$$

โดยที่ I_j : เป็นค่าของ element ตัวที่ j ของ input

w_{ij} : เป็นค่าของ weight ตัวที่ j ของโหนดที่ i

$\sum_j w1_j$: ผลรวมของ weight ที่มีค่าเป็นบวก

6.4 การปรับ weight

จากที่กล่าวในข้อ 6.1 และ 6.2 ค่าของ weight อยู่ในรูปของข้อมูลที่ผ่านการ normalized และ input vector อยู่ในรูป +1,-1 ดังนั้นการปรับ weight จะต้อง normalized ข้อมูลของ input vector ก่อน โดยใช้สมการที่ 5

$$W(t+1)=W(t)+\alpha(I/\sum I_i - W(t)) \quad \text{....(5)}$$

โดยที่ I . เป็น input vector

α learning rate ซึ่งมีค่าน้อยกว่า 1

$\sum I_i$ ผลรวมของค่า 1 ของ input vector

W : weight vector ที่เวลา t

การสอนโครงข่ายจะกำหนดให้ค่า α มีค่ามากเพื่อให้ค่าของ weight ปรับเข้าหา input ให้ได้รวดเร็ว จากนั้นจึงค่อยๆ ลดค่า α ลงเพื่อป้องกันไม่ให้ weight ออกข้างนอก centroid region

6.5 การใช้ค่า Threshold แก้ปัญหาการหาค่า weight เฉลี่ย

จากปัญหาการหาค่า weight เฉลี่ยในข้อ 5.2 งานวิจัยนี้จึงใช้วิธีการกำหนดค่า Threshold ของความเหมือนระหว่าง input กับ weight เพื่อแบ่งพื้นที่ของ win region ออกเป็นส่วนย่อย ซึ่งวิธีนี้ทำให้พื้นที่ของ win region มีขนาดแคบลง ซึ่งทำให้การหา weight เฉลี่ยทำได้ง่าย การสอนโครงข่าย ***ถ้า net-input ของ winner มีค่าน้อยกว่า Threshold ของความเหมือนระหว่าง weight และ input ให้ทำการ allocate node ของ weight ใหม่โดยให้มีค่า weight มีค่าเท่ากับ input-vector ที่ถูกกำหนดในขั้นตอน 6.1*** การกำหนดค่า Threshold ของแต่ละ input class มีค่าไม่เท่ากันขึ้นอยู่กับชนิดของ class (ในงานวิจัยนี้ class หมายถึง กลุ่มของ input ก ,ข เป็นต้น)

6.7 การแยก Train

รูปที่ 7 แสดงปัญหาการเกิด stuck weight ในขณะสอนโครงข่าย รูป 7 (a) W1 (ก) และ W2 (ฉ) เป็น weight ลักษณะตัวบาง เมื่อมี input I1 (ก) ที่มีลักษณะหนามาสอน W1 ถูกปรับดังแสดงในรูป 7 (b) ซึ่งเมื่อ input I2 (ฉ) เข้ามา train ทำให้ I2 ไ. ตกที่ W1 ซึ่งผิด

รูป 7 (a) แสดงการ train I1 (ก) เมื่อ I1 ที่มีลักษณะตัวหนาเข้า

รูป 7 (b) แสดงการ train I2 (ฉ) หลังการ train I2

เนื่องจาก CPN มี winner เพียงโหนด (node) เดียวที่ได้รับการปรับ weight ดังนั้นงานวิจัยนี้จึงทำการแยกสอน input pattern ที่ละ class เพื่อเป็นการบังคับให้ weight ตกในกลุ่มของ class ตัวเอง ซึ่งป้องกันปัญหาการเกิด stuck weight

6.8 สรุปขั้นตอนการสอนโครงข่าย

1.จัดแบ่ง input vector ออกเป็น class (class ก,ข,ค...) และกำหนดค่า Threshold ของความเหมือนให้กับแต่ละ class

2. กำหนดค่า α

3. ทำการแยกสอน input vector ทีละ class

3.1 เลือก input vector มาหนึ่งตัวเพื่อทำการกำหนดค่าเริ่มต้นของ weight ตามหัวข้อ 6.1

3.2 สุ่มเลือก input แปลงค่า 0,1 เป็น -1,1

3.3 หาค่า net input ตามหัวข้อ 6.3 เลือกโหนดที่มีค่า net input มากที่สุดเป็น winner

3.4 เปรียบเทียบค่า net input กับค่า Threshold ของความเหมือนของ class

ถ้า ค่า net input น้อยกว่าค่า Threshold ของความเหมือน

allocate weight ตามหัวข้อ 6.1

ไม่ใช่

ทำการปรับ weight ตามหัวข้อ 6.4

3.5.ทำให้ได้ตามจำนวนรอบที่ต้องการ

(จำนวนรอบในการ train แต่ละ class ต้องเท่ากัน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.ทำการทดสอบรวม

ถ้า เปอร์เซ็นต์ทดสอบรวมได้น้อยกว่าที่กำหนด

ให้ปรับค่า Threshold ของ class ที่มีการซ้อนทับของ pattern

ไม่ใช่ ให้ทำการลดค่า α ลงกลับทำข้อ 2

5.ทำจนกระทั่งได้ค่า weight เฉลี่ยที่ต้องการ

7. ผลการทดลอง

งานวิจัยนี้ได้ทดลองการจดจำตัวพิมพ์อักษรภาษาไทย โดยใช้ฟอนท์ (font) AngsanaUPC ขนาด 12 point อัตราส่วนในการ scan 300 dpi และ scale 100 % ตัวอักษรที่นำมาใช้ในการทดสอบประกอบด้วย พยัญชนะจำนวน 44 ตัว,เลขไทยและเลขภาษาอังกฤษ,เ,ไ,ใ,ๆ,า,ฤ,ฦ และ ไม้หันอากาศ จำนวนรวม 75 ตัวอักษร เมตริกของ input vector ที่ใช้ในการทดลองมีขนาด 28 x 38 โดยวางตัวอักษรที่เป็น input ที่ตำแหน่งชิดมุมบนซ้ายของ input metrix

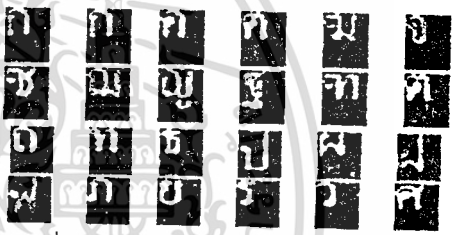
การทดลองนี้แบ่งตัวอักษรออกเป็น 4 ชุดคือ ตัวอักษรที่พิมพ์จากเครื่อง laser และตัวอักษรที่พิมพ์จากเครื่อง laser แล้วผ่านการถ่ายเอกสาร และความสว่างในการ scan อย่างละ 100 และ 50 % ตัวอย่างที่ใช้ในการสอนเลือกจากกลุ่มตัวอักษรที่นำมาทดสอบชุดตัวอักษรละ 20 แบบรวมแล้ว 1500 ตัวอักษร

การสอนเริ่มต้น กำหนดค่า weight เริ่มต้นโดยเลือกมาจากกลุ่มตัวอย่างที่ใช้ในการสอนกลุ่มละ 1 ตัว รวม 75 ตัว ซึ่งแทนด้วยโหนดใน competitive layer การสอนนี้ทำจนกระทั่งโครงข่ายสามารถแยกตัวอักษรได้ 100 % ตารางที่ 1 แสดงเปอร์เซ็นต์ความถูกต้องและจำนวนโหนดที่ได้จากการ train ทุก ๆ 4 รอบจำนวน 15 ครั้ง โดยกำหนดให้ค่าเริ่มต้นของ α เท่ากับ 0.8 แล้วทำการลดค่า α จนมีค่าเท่ากับ 0.2

จากการทดลองได้จำนวนโหนดทั้งสิ้น 143 โหนด โหนดของอักษรที่มีจำนวนโหนดมากจะอยู่ในกลุ่มของตัวอักษรคล้ายกัน เช่น ค 5 โหนด,ศ 7 โหนด,ด 4 โหนด,ต 7 โหนด ตารางที่ 1 ในแถวที่ 9 จะเห็นว่า input pattern สามารถถูกแยกได้ 100 เปอร์เซ็นต์ แต่ค่า weight ที่ได้ยังไม่ใช่ค่า weight เฉลี่ยที่แท้จริง เพราะเมื่อทำการสอนโครงข่ายต่อไปในครั้งที่ 10 และ 11 จะเห็นว่าเปอร์เซ็นต์ความถูกต้องลดลง ต่อมาทำการสอนโครงข่ายต่อในครั้งที่ 12 ถึง 15 ได้เปอร์เซ็นต์ที่ได้คงที่คือ 100 % ซึ่งแสดงว่า weight ที่ได้มีลักษณะเป็นค่าเฉลี่ย รูปที่ 8 แสดงตัวอย่างของ weight ที่ได้จากการทดลอง ซึ่งประกอบด้วย weight ของตัวอักษร ก,ก,ค,ศ,ฆ,จ,ช,ฌ,ญ,ฐ,ฑ,ด,ท,ธ,ป,ผ,ฝ,ฟ,ภ,ย,ร,ว,ศ

ครั้ง	ค่า α	จำนวนโหนด	จำนวน input ที่ถูกต้อง	% ความถูกต้องของ input ที่นำมา train
0	-	75	-	-
1	0.8	124	1454	96.93
2	0.8	132	1469	97.93
3	0.8	141	1473	98.19
4	0.6	142	1488	99.19
5	0.6	142	1492	98.46
6	0.6	142	1483	99.86
7	0.6	143	1489	99.26
8	0.4	143	1497	99.80
9	0.4	143	1500	100.00
10	0.4	143	1496	99.73
11	0.4	143	1499	99.93
12	0.2	143	1500	100.00
13	0.2	143	1500	100.00
14	0.2	143	1500	100.00
15	0.2	143	1500	100.00

ตารางที่ 1 เปอร์เซ็นต์การเรียนทุก ๆ 4 รอบ ใช้จำนวน input ที่มาสอน 1500 ตัว



รูปที่ 8 แสดงลักษณะของ weight ที่ได้จากการสอนโครงข่าย

จากการทดลองการกำหนดค่า Threshold ของความเหมือนระหว่าง weight และ input ให้กับกลุ่มของ input ถ้ากำหนดให้มีค่าสูงมากๆ จะทำให้จำนวนโหนดของ weight ที่ allocate ในขั้นตอนของการเรียนรู้มีจำนวนมาก เนื่องจากพื้นที่ของ win region มีขนาดแคบมาก ซึ่งทำให้จำนวนของโหนดใน competitive layer อาจมีจำนวนมากเกินไป ดังนั้นการกำหนด Threshold ของความเหมือนของแต่ละ class จึงต้องมีการพิจารณา การกำหนดค่า Threshold นี้ควรให้มีค่าสูง ๆ กรณีกลุ่มของ input มีรูปแบบคล้าย ๆ กันเช่น ค,ศ,ด,ต เพื่อให้สามารถแยกตัวอักษรหรือจดจำตัวอักษรได้เร็วขึ้นเพราะ win region ของตัวอักษรกลุ่มนี้มีค่าใกล้เคียงกันมาก ส่วนตัวอักษรของกลุ่มที่ไม่คล้ายกลุ่มอื่นเช่น เลข ฐ,๓,๖ ก็ควรมีค่าน้อย ๆ เพราะกลุ่มของตัวอักษรนี้มีลักษณะแตกต่างจากกลุ่มอื่น เพื่อป้องกันไม่ให้ weight allocate มากเกินไป

ชนิดกระดาษที่ทดสอบ (%ความชัดว่าง)	จำนวนตัวอักษรที่ถูกต้องที่ทดสอบ	เปอร์เซ็นต์ความถูกต้องที่ทดสอบ
พิมพ์ laser (100 %)	14918	99.45
พิมพ์ laser (50 %)	14303	95.35
ถ่ายเอกสาร (100%)	14902	99.34
ถ่ายเอกสาร (50%)	14036	93.57

ตารางที่ 2 แสดงผลการทดสอบการแยกตัวอักษรภาษาไทยชุดละ 15000 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 9 แสดงตัวอย่าง input ที่สามารถจดจำได้

รูปที่ 10 แสดงตัวอย่าง input ที่ไม่สามารถจดจำ

ตารางที่ 2 แสดงผลการทดสอบการแยกตัวอักษร โดยทดสอบกับกระดาษที่พิมพ์จากเครื่อง laser และกระดาษถ่ายเอกสารที่ความสว่าง 100 % และ 50 % ชุดละ 15,000 ตัว รูปที่ 9 แสดงตัวอย่างตัวอักษรที่สามารถแยกชนิดตัวอักษรได้ รูปที่ 10 แสดงตัวอย่างที่ไม่สามารถแยกชนิดตัวอักษรได้ (ตัวอักษรในวงเล็บแสดงโหนดของ weight ที่ input pattern ตก)

8. สรุป

การนำ CPN มาใช้ในการจดจำตัวอักษรนี้ใช้วิธีการสอนตัวอย่าง input ของตัวอักษรที่ได้จากการ scan ภาพ ลักษณะการเรียนรู้ของ CPN จะพยามสร้าง weight เฉลี่ยให้กับ input vector ที่นำใช้ในการสอนโครงข่าย เวลาที่ใช้ในการเรียนรู้ของ CPN ใช้เวลาน้อยมาก จากการทดลองที่กล่าวข้างต้นใช้เวลาการเรียนรู้ประมาณ 30 นาที (บนเครื่อง HP 9000/827) จากตัวอย่างข้อมูล 1500 ตัว

แม้ว่างานวิจัยนี้ได้นำวิธีการกำหนดค่า Threshold มาช่วยลอนโครงข่ายเพื่อให้สามารถหาค่า weight เฉลี่ยได้เร็วขึ้น แต่ในการกำหนดค่า Threshold ให้กับ class ต่าง ๆ นี้ต้องกำหนดให้เหมาะสมเพราะว่า ถ้าหากกำหนดค่า Threshold ให้มีค่าสูง ๆ แล้วก็อาจทำให้โหนดของ weight มีค่ามากเกินเพราะ win region มีขนาดแคบ

เนื่องจากการเรียนของ CPN จะพยามสร้าง look-up table ของ weight เฉลี่ยให้กับ input vector ที่นำใช้ในการสอนโครงข่าย ถ้าหากนำ input ที่มีขนาดใหญ่กว่าหรือตำแหน่งภาพตัวอักษรเลื่อนจาก input ที่นำมาใช้ในการสอนก็อาจทำให้โครงไม่สามารถจดจำ input pattern นั้นได้

9.หนังสืออ้างอิง

[1] Robert Hecht-Nielsen,*Neurocomputing*, Addison-Wesley , Reading, MA, 1990.

[2] James A.Freeman and David,"The Counterpropation Network." *Neural Networks Algorithms Applications and Programming Techniques*, Addison-Wesley, pp. 213-262,1991.

[3] Laurene Fausett,"Neural Networks Based on Competition" *Fundamentals of Neural Networks Architectures,Algorithms and Applications*, Prentice-Hall, pp. 156-217, 1994.

[4] Yon-Han Pao,"Self-Organizing Nes for Pattern Recognition." Adaptive Pattern Recognition and Neural Networks , Addison-Wesly, pp. 171-175, 1989.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อผู้เขียน	นายอภิรักษ์ จิรายุสกุล
วันเดือนปีเกิด	20 พฤษภาคม พ.ศ.2513
สถานที่เกิด	จังหวัดกาญจนบุรี
วุฒิการศึกษาระดับปริญญาตรี	วท.บ.(วิทยาการคอมพิวเตอร์)
สถานที่สำเร็จการศึกษา	คณะวิทยาศาสตร์ มหาวิทยาลัยรามคำแหง
ปีที่สำเร็จการศึกษา	ปีการศึกษา 2534
ผลงานวิชาการ	ประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 18 เรื่อง "การรู้จำตัวพิมพ์อักษรไทยโดยใช้ Counterpropagation Neural Network (CPN)"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้