

โครงข่ายประสาทเทียมสำหรับการแปลท่ามือหนึ่งในภาษามือไทย

ARTIFICIAL NEURAL NETWORK FOR THAI STATIC SIGN
LANGUAGE TRANSLATION



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาทฤษฎีและปฏิบัติของวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2548

ISBN 974-15-1661-4

โครงข่ายประสาทเทียมสำหรับการแปลท่ามือนิ่งในภาษามือไทย

ARTIFICIAL NEURAL NETWORK FOR THAI STATIC SIGN

LANGUAGE TRANSLATION



กำพล วาสนาพงศ์พันธุ์

KUMPHOL WASANAPONGPAN

เลขหมู่.....
เลขทะเบียน..... 60846
วัน,เดือน,ปี... - 6 ป.ศ. 2549

b.....
i.....

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2548

ISBN 974-15-1661-4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ARTIFICIAL NEURAL NETWORK FOR THAI STATIC SIGN
LANGUAGE TRANSLATION**



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF SCIENCE IN INFORMATION TECHNOLOGY
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2005

ISBN 974-15-1661-4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2005

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	โครงข่ายประสาทเทียมสำหรับการแปลท่ามือนิ่งในภาษามือไทย
นักศึกษา	นายกำพล วาสนาพงศ์พันธุ์
รหัสประจำตัว	43067019
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	วิทยาการสารสนเทศ
พ.ศ.	2548
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ.ดร.นพพร โชติศักดิ์

บทคัดย่อ

วิทยานิพนธ์ฉบับนี้ ได้นำเสนอผลการวิจัยการประยุกต์ใช้โครงข่ายประสาทเทียมสำหรับการแปลท่ามือนิ่งในภาษามือไทย (Thai Sign Language) ส่วนแรกจะเป็นผลการศึกษาเชิงเปรียบเทียบวิธีการที่ใช้สำหรับการแปลท่ามือนิ่งสามวิธี คือ Bayesian Estimator และโครงข่ายประสาทเทียม (Neural Network) แบบ Multi-Layer Perceptron Neural Network (MLP) และ Radial Basis Function Neural Network (RBF) ในประเด็นของความถูกต้องในการรู้จำและความซับซ้อนในการคำนวณขณะแปลความหมาย จากผลการทดลองพบว่า วิธีการที่เปรียบเทียบให้ค่าความถูกต้องที่ใกล้เคียงกัน ซึ่งวิธีการแบบ Bayesian Estimator มีความถูกต้องดีที่สุด ขณะที่โครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron มีความซับซ้อนในการคำนวณขณะแปลความหมายน้อยที่สุด นอกจากนี้ วิทยานิพนธ์ฉบับนี้ ยังได้นำเสนอผลการพัฒนาโครงข่ายประสาทเทียมสำหรับแปลความหมายของท่ามือบอกขนาดหรือปริมาณ (Quantitative Posture) ในภาษามือไทย ซึ่งเรียกโครงข่ายนี้ว่า Quantitative-classification Multi-Layer Perceptron (QMLP) วิธีการที่นำเสนอสามารถลดปริมาณข้อมูลที่โครงข่ายประสาทเทียมต้องการในการเรียนรู้ ยกตัวอย่างเช่น โครงข่ายแบบ QMLP สามารถแปลความหมายของขนาดหรือปริมาณได้ห้าระดับ โดยใช้ข้อมูลในการเรียนรู้เพียงสามระดับ จากผลการทดลองพบว่า โครงข่ายแบบ QMLP และ SMLP ให้ค่าความถูกต้องใกล้เคียงกัน

Thesis	Artificial Neural Network for Thai Static Sign Language Translation
Student	Mr. Kumphol Wasanapongpan
Student ID	43067019
Degree	Master of Science
Programme	Information Science
Year	2005
Thesis Advisor	Assoc.Prof.Dr.Nopporn Chotikakamthorn

ABSTARCT

This thesis presents research result on the application of an artificial neural network for automated static hand posture translation in Thai Sign Language. The first part is a comparative study of the three methods for static hand postures translation. The three methods are Bayesian Estimator, and two neural networks. The networks used are Multi-Layer Perceptron Neural Network (MLP) and Radial Basis Function Neural Network (RBF). Comparison was made in terms of accuracy and recognition-stage computational complexity. From experimental result, it was found that these three methods gave comparable accuracy level. The Bayesian Estimator produced a slightly better accuracy while the Multi-Layer Perceptron required lowest translation computation complexity. In addition this thesis presents a newly developed artificial neural network for quantitative posture translation in Thai Sign Language. The network is referred to as Quantitative-classification Multi-Layer Perceptron (QMLP). The proposed method can reduce the amount of training data that neural network required to learning. For example, the QMLP method can translate five quantitative levels by using training data of three different quantitative levels. From experimental result, it was found that QMLP and SMLP methods gave comparable accuracy level.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้ สามารถสำเร็จลุล่วงไปด้วยดี ด้วยการให้คำปรึกษา และแนะนำแนวทางจาก รศ.ดร.นพพร โชติกคำธร ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ ผู้วิจัยขอกราบขอบพระคุณเป็นอย่างสูงสำหรับความอนุเคราะห์จากท่าน

ขอกราบขอบพระคุณ บิดา มารดา และ พี่ ๆ ของนักวิจัยที่ให้การสนับสนุนและช่วยเหลือในด้านต่าง ๆ อีกทั้งเป็นกำลังใจให้กับผู้วิจัยเสมอมา

ขอขอบคุณบุคคลต่อไปนี้ ที่อาสาช่วยเหลือในการเก็บข้อมูลสำหรับใช้ในการทดลอง ซึ่งได้แก่ เกียรติ, สิทธิ, เป้ว, แหวว, หนูย, กบ, ก้อย, นิด และนุ้ย

อนึ่ง งานวิจัยที่นำเสนอในวิทยานิพนธ์ฉบับนี้นั้น ได้ปฏิบัติภายในห้องปฏิบัติการ Multimedia and Virtual Research (MVLAB) ของสำนักวิจัยการสื่อสารและเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

คุณค่าและประโยชน์ที่พึงได้รับจากวิทยานิพนธ์ฉบับนี้ ขอมอบแด่ผู้มีพระคุณทุกท่าน

กำพล วาสนาพงศ์พันธุ์

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการศึกษา.....	2
1.3 สมมติฐานของการศึกษา.....	2
1.4 ทฤษฎีหรือแนวความคิดที่เกี่ยวข้องกับงานวิจัย.....	3
1.5 ขอบเขตของงานวิจัย.....	3
1.6 ขั้นตอนการศึกษา.....	3
1.7 โครงสร้างวิทยานิพนธ์.....	4
บทที่ 2 งานวิจัยที่เกี่ยวข้อง.....	5
2.1 ระบบแปลภาษามืออัตโนมัติ.....	5
2.1.1 งานวิจัยของ C. Shahabi, L. Kaghazian, S. Metha, A. Ghoting, G. Shanbhag และ M. McLaughlin.....	5
2.1.2 งานวิจัยของ Worawit Werapan และ Nopporn Chotikakamthorn.....	5
2.1.3 งานวิจัยของ Manjula B. Waldron และ Soowon Kim.....	6
2.1.4 งานวิจัยของ S. Sidney Fels และ Geoffrey E. Hinton.....	7
2.1.5 งานวิจัยของ Peter Vamplew.....	8
บทที่ 3 หลักการรู้จำรูปแบบ.....	10
3.1 Bayesian Estimator.....	10
3.2 โครงข่ายประสาทเทียม (Neural Network).....	13
3.2.1 Multi-Layer Perceptron Neural Network (MLP).....	13

สารบัญ (ต่อ)

หน้า

3.2.2 Radial Basis Function Neural Network (MLP)	19
บทที่ 4 การพัฒนาโครงข่ายประสาทเทียมสำหรับการรู้จำท่ามือบอกขนาดหรือปริมาณ	26
4.1 แนวคิดเบื้องต้น	26
4.2 การพัฒนาโครงข่ายประสาทเทียมสำหรับการรู้จำท่ามือบอกขนาดหรือปริมาณ	27
4.2.1 โครงสร้างของโครงข่าย	27
4.2.1.1 Standard Multi-Layer Perceptron (SMLP)	27
4.2.1.2 Quantitative-classification Multi-Layer Perceptron-Type I (QMLP-I) ...	29
4.2.1.3 Quantitative-classification Multi-Layer Perceptron-Type II (QMLP-II).	30
4.2.2 การกำหนดค่าผลลัพธ์ที่ต้องการ (Desired Output) ให้กับข้อมูล	31
4.2.2.1 Standard Multi-Layer Perceptron (SMLP)	31
4.2.2.2 Quantitative-classification Multi-Layer Perceptron-Type I (QMLP-I) ...	32
4.2.2.3 Quantitative-classification Multi-Layer Perceptron-Type II (QMLP-II).	34
4.2.3 การเรียนรู้ของ โครงข่าย	35
4.2.4 การรู้จำของ โครงข่าย	36
4.2.2.1 Standard Multi-Layer Perceptron (SMLP)	36
4.2.2.2 Quantitative-classification Multi-Layer Perceptron-Type I (QMLP-I) ...	36
4.2.2.3 Quantitative-classification Multi-Layer Perceptron-Type II (QMLP-II).	40
บทที่ 5 อุปกรณ์และข้อมูลที่ใช้ในการทดลอง	43
5.1 อุปกรณ์ที่ใช้ในการทดลอง	43
5.1 ข้อมูลที่ใช้ในการทดลอง	44
บทที่ 6 วิธีการทดลองและผลการทดลอง	47
6.1 การเปรียบเทียบวิธีการสำหรับรู้จำท่ามือ	47
6.1.1 วิธีการทดลอง	47
6.1.2 ผลการทดลอง	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
6.1.2.1 การเปรียบเทียบความถูกต้อง.....	48
6.1.2.2 การเปรียบเทียบความความซับซ้อนในการคำนวณขณะแปลความหมาย.....	52
6.1.3 สรุปผลการทดลอง	54
6.2 การพัฒนาโครงข่ายประสาทเทียมสำหรับรู้จำท่ามือบอกขนาดหรือปริมาณ.....	54
6.1.1 วิธีการทดลอง	54
6.1.2 ผลการทดลอง.....	55
6.1.3 สรุปผลการทดลอง	61
บทที่ 7 สรุปผลการทดลองและข้อเสนอแนะ.....	62
เอกสารอ้างอิง	63
ภาคผนวก	65
ภาคผนวก ก. สมการที่ใช้สำหรับคำนวณความซับซ้อนขณะแปลความหมาย.....	66
ภาคผนวก ข. บทความและผลงานวิจัยที่ได้รับการตีพิมพ์	74
ประวัติผู้เขียน.....	87

สารบัญตาราง

ตารางที่	หน้า
6.1 แสดงค่าความถูกต้องในกรณีของ Signer Dependence.....	49
6.2 แสดงค่าความถูกต้องในกรณีของ Signer Independence	50
6.3 แสดงความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการที่ใช้ในการเปรียบเทียบ ในกรณี Signer Dependence	53
6.4 แสดงความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการที่ใช้ในการเปรียบเทียบ ในกรณี Signer Independence.....	54
6.5 เปรียบเทียบค่าความถูกต้องของการใช้โครงข่ายประสาทเทียมแบบ SMLP และ โครงข่าย ประสาทเทียมแบบ QMLP ในการรู้จำท่ามือบอกขนาดหรือปริมาณในกรณีของ Signer Dependence	55
6.6 เปรียบเทียบค่าความถูกต้องของการใช้โครงข่ายประสาทเทียมแบบ SMLP และ โครงข่าย ประสาทเทียมแบบ QMLP ในการรู้จำท่ามือบอกขนาดหรือปริมาณในกรณี Signer Independence	56
6.7 แสดงความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการที่ใช้ในการเปรียบเทียบ กรณี ที่ทำมือบอกขนาดหรือปริมาณแต่ละท่าต้องการรู้จำขนาดหรือปริมาณ 5 ระดับ	57
6.8 แสดงค่าความถูกต้องกรณีที่กำหนดค่าองศาของ โครงข่ายประสาทเทียมแบบ QMLP-I เท่ากับ 90 63 และ 55 องศา เปรียบเทียบกับ โครงข่ายประสาทเทียมแบบ QMLP-II ใน การรู้จำท่ามือบอกขนาดหรือปริมาณในกรณีของ Signer Dependence	60
6.9 แสดงค่าความถูกต้องกรณีที่กำหนดค่าองศาของ โครงข่ายประสาทเทียมแบบ QMLP-I เท่ากับ 90 63 และ 55 องศา เปรียบเทียบกับ โครงข่ายประสาทเทียมแบบ QMLP-II ใน การรู้จำท่ามือบอกขนาดหรือปริมาณในกรณีของ Signer Independence	60
ก.1 สมการที่ใช้สำหรับคำนวณความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการ ที่เปรียบเทียบ	72
ก.2 แสดงความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการที่ใช้ในการเปรียบเทียบใน กรณี Signer Dependence.....	72
ก.3 แสดงความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการที่ใช้ในการเปรียบเทียบใน กรณี Signer Independence	73

สารบัญรูป

รูปที่	หน้า
2.1 Thai Sign Language Translation System	6
2.2 ระบบแปลภาษาของ Manjala B. Waldron และ Soowon Kim	7
2.3 ระบบ SLARTI	8
3.1 ขั้นตอนการทำงานของวิธีการแบบ Bayesian Estimator.....	11
3.2 โครงสร้างของโครงข่ายประสาทเทียมแบบ MLP	13
3.3 โค้ดแกรมแสดงขั้นตอนการเรียนรู้ของโครงข่ายประสาทเทียมแบบ MLP	16
3.4 โค้ดแกรมแสดงขั้นตอนการรู้จำของโครงข่ายประสาทเทียมแบบ MLP	18
3.5 โครงสร้างของโครงข่ายประสาทเทียมแบบ Radial Basis Function Network	19
3.6 โค้ดแกรมแสดงขั้นตอนการเรียนรู้ของโครงข่ายประสาทเทียมแบบ RBF	24
3.7 โค้ดแกรมแสดงขั้นตอนการรู้จำของโครงข่ายประสาทเทียมแบบ RBF	25
4.1 ทำมือบอกขนาดหรือปริมาณที่มีความหมายว่า “ความหนาของวัตถุ”	26
4.2 โครงข่ายประสาทเทียมแบบ SMLP สำหรับรู้จำทำมือบอกขนาดหรือปริมาณ กรณีที่ทำมือบอก ขนาดหรือปริมาณต้องการรู้จำขนาดหรือปริมาณ 3 ระดับ.....	27
4.3 โครงข่ายประสาทเทียมแบบ SMLP สำหรับรู้จำทำมือบอกขนาดหรือปริมาณ กรณีที่ทำมือบอก ขนาดหรือปริมาณต้องการรู้จำขนาดหรือปริมาณ 5 ระดับ.....	28
4.4 โครงข่ายประสาทเทียมแบบ QMLP-I สำหรับรู้จำทำมือบอกขนาดหรือปริมาณ	29
4.5 โครงข่ายประสาทเทียมแบบ QMLP-II สำหรับรู้จำทำมือบอกขนาดหรือปริมาณ	30
4.6 การกำหนดค่าผลลัพธ์ที่ต้องการให้กับข้อมูลของทำมือบอกขนาดหรือปริมาณของวิธีการ แบบ QMLP-I	33
4.7 การกำหนดระดับขนาดให้กับทำมือบอกขนาดหรือปริมาณของโครงข่ายแบบ QMLP-I.....	33
4.8 การกำหนดระดับขนาดให้กับทำมือบอกขนาดหรือปริมาณของโครงข่ายแบบ QMLP-II.....	35
4.9 โค้ดแกรมแสดงขั้นตอนการรู้จำทำมือบอกขนาดหรือปริมาณของโครงข่ายแบบ SMLP.....	36
4.10 แสดงการรู้จำความหมายของโครงข่ายประสาทเทียมแบบ QMLP-I.....	37
4.11 แสดงการรู้จำขนาดหรือปริมาณของโครงข่ายประสาทเทียมแบบ QMLP-I.....	38
4.12 โค้ดแกรมแสดงขั้นตอนการรู้จำข้อมูลของโครงข่ายแบบ QMLP-I	39
4.13 แสดงการรู้จำความหมายของโครงข่ายประสาทเทียมแบบ QMLP-II.....	40
4.14 แสดงการรู้จำขนาดหรือปริมาณของโครงข่ายประสาทเทียมแบบ QMLP-II.....	41
4.15 โค้ดแกรมแสดงขั้นตอนการรู้จำข้อมูลของโครงข่ายแบบ QMLP-II	42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
5.1 ถู่มืออิเล็กทรอนิกส์	43
5.2 ตัวอย่างท่ามือที่มีความหมายว่า “ลืม” ในระบบภาษามือไทย	44
5.3 ตัวอย่างท่ามือพื้นฐานของภาษามือไทย	45
5.4 ตัวอย่างท่ามือบอกขนาดหรือปริมาณ (Quantitative Posture).....	45
6.1 Histogram และเส้นโค้งปกติของข้อมูลที่ใช้ในการทดลอง	48
6.2 แสดงค่าความถูกต้องของวิธีการที่ใช้ในการเปรียบเทียบในกรณี Signer Dependence	49
6.3 แสดงค่าความถูกต้องของวิธีการที่ใช้ในการเปรียบเทียบในกรณี Signer Independence.....	50
6.4 แสดงค่า RMSE ของแต่ละท่ามือ กรณี Signer Dependence	51
6.5 แสดงขนาดความกว้างของแต่ละช่วงชั้นของโครงข่ายประสาทเทียมแบบ QMLP-II.....	58
6.6 แสดงขนาดความกว้างของแต่ละช่วงชั้นของโครงข่ายประสาทเทียมแบบ QMLP-I	59



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในการสื่อสารของมนุษย์นั้น สามารถจำแนกออกเป็น 2 วิธี คือ การสื่อสารด้วยคำพูด หรือที่เรียกว่า “วจนภาษา” และการสื่อสารด้วยท่าทางหรือสัญลักษณ์ ที่เรียกว่า “อวจนภาษา” ซึ่งภาษามือนั้น จัดอยู่ในอวจนภาษา กลุ่มคนที่ใช้ภาษามือส่วนใหญ่จะเป็นบุคคลที่มีความพิการทางการได้ยิน หรือ คนหูหนวก จึงทำให้บุคคลกลุ่มนี้มีปัญหาในการติดต่อสื่อสารกับบุคคลปกติทั่วไป เนื่องจากบุคคลปกติจะไม่มีความรู้เกี่ยวกับภาษามือ ในปัจจุบันได้มีหน่วยงานหรือสถาบันต่าง ๆ ได้ทำการศึกษา วิจัย ถอดค้นประดิษฐ์อุปกรณ์ต่าง ๆ ที่ช่วยให้บุคคลที่มีความพิการทางการได้ยินสามารถที่จะติดต่อสื่อสารกับบุคคลทั่วไปได้ ไม่ว่าจะเป็นพจนานุกรมภาษามืออิเล็กทรอนิกส์ หรือ วัสดุสื่อการสอน เป็นต้น ซึ่งระบบหนึ่งที่ได้รับคามนิยมเป็นอย่างมาก ที่เรียกว่า ระบบแปลภาษามือ กล่าวคือ เป็นระบบที่ใช้ในการแปลภาษามือที่ทำโดยผู้ที่มีความพิการทางการได้ยินให้ออกมาในรูปของสื่อที่บุคคลปกติสามารถเข้าใจได้ ไม่ว่าจะออกมาในรูปของเสียงหรือตัวอักษรก็ตาม

จากการศึกษางานวิจัยที่เกี่ยวข้องกับระบบรู้จำภาษามือ พบว่าภายในระบบจำเป็นต้องมีวิธีการที่ใช้ในการรู้จำข้อมูลที่ถูกส่งเข้าไปในระบบ เพื่อรู้ว่าข้อมูลที่ส่งเข้าไบนั้นมีความหมายอย่างไร ดังนั้นจึงมีความจำเป็นอย่างยิ่งที่ต้องมีการศึกษาหาวิธีการที่เหมาะสม เพื่อที่จะนำไปใช้ในการรู้จำข้อมูลของท่ามือในระบบแปลภาษามือ

ภาษามือ เป็นภาษาที่มีการใช้งานในทุกประเทศ บางประเทศก็มีภาษามือของตนเอง แต่บางประเทศก็มีการอ้างอิงภาษามือของประเทศอื่น แต่อย่างไรก็ตามภาษามือทุกภาษาก็มีองค์ประกอบหลักที่คล้ายคลึงกัน ดังนี้

1. ท่ามือหรือรูปมือ (Posture or Hand Shape)
2. การหมุนหรือการหักข้อมือ (Orientation of Palm)
3. ตำแหน่งของมือที่สัมพันธ์กับร่างกาย (Position and Location of Hand)
4. การเคลื่อนไหวของแขน (Hand Movement)

ซึ่งองค์ประกอบทั้ง 4 ส่วนนั้นล้วนมีความสำคัญต่อความหมายในระบบภาษามือทั้งสิ้น

จากการศึกษาท่ามือภาษาไทย (Thai Sign Language) พบว่า มีท่ามือบางท่าเป็นท่ามือบอกขนาดหรือปริมาณ กล่าวคือ เป็นท่ามือที่มีความหมายในการบอกขนาดหรือปริมาณที่เฉพาะเจาะจงว่าวัตถุหรือสิ่งของมีขนาดหรือมีปริมาณเท่าใด

ซึ่งทำมือบอกขนาดหรือปริมาณเหล่านี้ ทำให้เกิดปัญหาในเรื่องของจำนวนข้อมูลของทำมือบอกขนาดหรือปริมาณที่ใช้ในการเรียนรู้ กล่าวคือ ต้องใช้ข้อมูลในระดับต่าง ๆ ทุกระดับที่ต้องการให้สามารถจำแนกได้ ซึ่งวิธีการที่ใช้ในการรู้จำทำมือบอกขนาดหรือปริมาณ จำเป็นต้องสามารถแยกแยะทำมือบอกขนาดหรือปริมาณเหล่านี้ได้ การใช้โครงข่ายประสาทเทียมมาตรฐานแบบ multi-layer perceptron ทำให้ต้องอาศัยข้อมูลในการเรียนรู้ที่แทนทำมือบอกขนาดหรือปริมาณในระดับต่าง ๆ ทุกระดับที่ต้องการให้สามารถจำแนกได้ ยกตัวอย่างเช่น ถ้าต้องการให้โครงข่ายสามารถจำแนกทำมือบอกขนาดหรือบอกปริมาณทั้งหมด 5 ระดับ ก็ต้องใช้ข้อมูลของระดับทั้ง 5 ระดับ ในการเรียนรู้ของโครงข่าย ดังนั้น ในงานวิจัยนี้จึงได้เสนอวิธีการที่จะช่วยลดปัญหาดังกล่าว โดยอาศัยข้อมูลในการเรียนรู้จากทำมือบอกระดับขนาดหรือปริมาณที่น้อยกว่าการใช้โครงข่ายประสาทเทียมมาตรฐานแบบ multi-layer perceptron

1.2 วัตถุประสงค์ของการศึกษา

- เพื่อศึกษาเปรียบเทียบประสิทธิภาพในการรู้จำทำมือภาษาไทย ระหว่างวิธีการทางสถิติ คือ Bayesian Estimator กับโครงข่ายประสาทเทียม (Neural Network) ซึ่งในที่นี้จะทำการศึกษาโครงข่ายประสาทเทียม 2 แบบ คือ Multi-Layer Perceptron Neural Network และ Radial Basis Function Neural Network ตลอดจนศึกษาปัจจัยที่มีผลต่อความถูกต้อง ในการรู้จำทำมือของโครงข่ายประสาทเทียม
- เพื่อพัฒนาโครงข่ายประสาทเทียม สำหรับรู้จำทำมือภาษาไทยที่เป็นทำมือบอกขนาดหรือปริมาณ (Quantitative Posture)

1.3 สมมติฐานของการศึกษา

- จำนวนโหนด (Node) ในชั้นซ่อน (Hidden Layer) และจำนวนข้อมูลที่ใช้ในการเรียนรู้ของโครงข่ายประสาทเทียมมีผลต่อประสิทธิภาพในการรู้จำของโครงข่ายประสาทเทียม กล่าวคือ ถ้าจำนวนโหนดในชั้นซ่อนและจำนวนข้อมูลที่ใช้ในการเรียนรู้มีจำนวนเพียงพอ ก็จะทำให้ประสิทธิภาพในการรู้จำของโครงข่ายประสาทเทียมมีประสิทธิภาพที่ใกล้เคียงกับวิธีการแบบ Bayesian Estimator อันเป็นผลมาจากคุณสมบัติความเป็น universal classifier/estimator ของโครงข่ายประสาทเทียมที่ศึกษา
- โครงข่ายประสาทเทียมมีคุณสมบัติที่เรียกว่า Interpolation กล่าวคือ โครงข่ายประสาทเทียมสามารถจะประมาณค่าที่อยู่ระหว่างค่า 2 ค่าได้ จากคุณสมบัตินี้ จึงสามารถประยุกต์โครงข่ายประสาทเทียมเพื่อใช้ในการรู้จำทำมือภาษาไทยที่เป็นทำมือบอกขนาดหรือปริมาณ (Quantitative

Posture) กล่าวคือ โครงข่ายประสาทเทียมสามารถที่จะรู้จำขนาด (Quantitative Level) ของท่ามือได้มากกว่าระดับขนาดของท่ามือในขั้นตอนการเรียนรู้

1.4 ทฤษฎีหรือแนวความคิดที่เกี่ยวข้องกับงานวิจัย

ประสิทธิภาพของระบบแปลท่ามือขึ้นอยู่กับองค์ประกอบหลายส่วน ส่วนหนึ่งที่มีความสำคัญไม่น้อยไปกว่าส่วนอื่น คือ ส่วนที่ใช้ในการรู้จำข้อมูล ซึ่งถ้าระบบแปลท่ามือ มีวิธีการรู้จำที่เหมาะสม ไม่ว่าจะเป็นประเด็นในเรื่องของความถูกต้อง หรือความซับซ้อนในการคำนวณขณะแปลความหมาย ซึ่งในปัจจุบันมีวิธีการรู้จำที่เป็นที่นิยมอยู่ 2 วิธี คือ วิธีการทางด้านสถิติ เช่น Bayesian Estimator และวิธีการแบบโครงข่ายประสาทเทียม แต่ละวิธีการก็จะมีเหมาะสมกับงานที่แตกต่างกันไป

1.5 ขอบเขตของงานวิจัย

1. ศึกษาระบบภาษามือ และองค์ประกอบของภาษามือไทย
2. ศึกษาวิธีการที่ใช้ในการรู้จำ 2 ทฤษฎี คือ ทฤษฎีที่อ้างอิงทางด้านสถิติ คือ วิธีการ Bayesian Estimator และ โครงข่ายประสาทเทียม ซึ่งจะศึกษาเพียง 2 แบบ คือ multi-layer perceptron neural network และ radial basis function neural network
3. ทำการทดลองเปรียบเทียบวิธีการรู้จำท่ามือภาษาไทย ในประเด็นของความถูกต้อง และความซับซ้อนในการคำนวณขณะแปลความหมาย
4. พัฒนาโครงข่ายประสาทเทียมสำหรับรู้จำท่ามือบอกขนาดหรือปริมาณ

1.6 ขั้นตอนการศึกษา

1. กำหนดหัวข้อ เป้าหมาย วัตถุประสงค์ และ ขอบเขตของการทำวิทยานิพนธ์
2. ศึกษาทฤษฎี และ หลักการพื้นฐานที่เกี่ยวข้องกับงานวิจัย
3. ศึกษาวิธีการต่างๆ ที่มีอยู่ แนวความคิด หลักการ ข้อดี และ ข้อบกพร่องของแต่ละวิธีการ
4. ศึกษาเปรียบเทียบวิธีการในการรู้จำท่ามือภาษาไทย และประยุกต์โครงข่ายประสาทเทียมในการรู้จำท่ามือบอกขนาดหรือบอกปริมาณ
5. ทำการทดลอง ปรับปรุง และ สรุปผล
6. จัดทำเอกสารประกอบวิทยานิพนธ์

1.7 โครงสร้างวิทยานิพนธ์

บทที่ 1 กล่าวถึงความเป็นมา และ ความสำคัญของปัญหาที่นำมาสู่การศึกษาเพื่อทำวิจัย กล่าวถึง หลักการ และ ทฤษฎีต่างๆ ภายใต้สมมุติฐานที่กำหนด โดยมีจุดมุ่งหมาย ขอบเขต และขั้นตอนในการศึกษาระบุไว้อย่างชัดเจน

บทที่ 2 กล่าวถึงหลักการ และ งานวิจัยที่เกี่ยวข้องกับการรู้จำท่ามือที่ปรากฏอยู่ในงานวิจัยต่าง ๆ

บทที่ 3 กล่าวถึงทฤษฎีที่ใช้ในการรู้จำข้อมูล

บทที่ 4 กล่าวถึงการพัฒนาโครงข่ายประสาทเทียมสำหรับรู้จำท่ามือบอกขนาดหรือปริมาณ

บทที่ 5 กล่าวถึงอุปกรณ์และข้อมูลสำหรับการทดลอง

บทที่ 6 กล่าวถึงผลการทดลองการศึกษาเปรียบเทียบวิธีการในการรู้จำท่ามือ และการพัฒนาโครงข่ายประสาทเทียมสำหรับรู้จำท่ามือบอกขนาดหรือปริมาณ

บทที่ 7 กล่าวถึงบทสรุปของการทดลองและข้อเสนอแนะ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

งานวิจัยที่เกี่ยวข้อง

2.1 ระบบแปลภาษามืออัตโนมัติ

งานวิจัยฉบับนี้จะเกี่ยวข้องกับงานวิจัยที่ศึกษาเกี่ยวกับระบบการแปลภาษามือแบบอัตโนมัติ (Automated Sign Language Translation System) ซึ่งเป็นการแปลภาษามือของประเทศต่าง ๆ โดยการใช้ถุงมืออิเล็กทรอนิกส์เชื่อมต่อกับคอมพิวเตอร์ แล้วแปลภาษามือให้ออกมาในรูปของเสียงหรือตัวอักษร ซึ่งงานวิจัยที่เกี่ยวข้อง สามารถสรุปได้ ดังนี้

2.1.1 งานวิจัยของ C. Shahabi, L. Kaghazian, S. Mehta, A. Ghoting, G. Shanbhag และ M. McLaughlin [1]

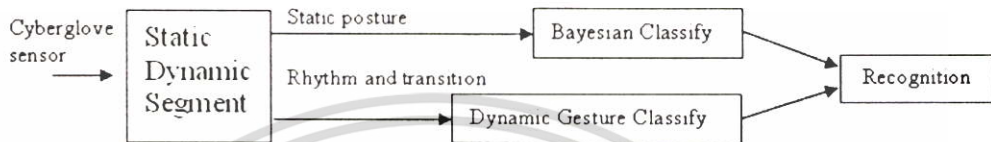
ได้ทำการศึกษาเชิงเปรียบเทียบวิธีการรู้จำท่ามือภาษาอเมริกัน (American Sign Language, ASL) ซึ่งวิธีการรู้จำที่นำมาเปรียบเทียบ ทั้งหมด 3 วิธี คือ Bayesian Classification, C4.5 Decision Tree และ Neural Network แบบ Multi-Layer Perceptron ที่ใช้ Back-Propagation algorithm ในการเรียนรู้โครงข่าย การเก็บข้อมูลจะใช้ถุงมืออิเล็กทรอนิกส์ ที่เรียกว่า CyberGlove ในการเก็บข้อมูลจำนวนของท่ามือที่ใช้ในการศึกษามีทั้งหมด 10 ท่า คือ ท่าที่มีความหมายเป็นตัวอักษร A – I และ L ซึ่งท่ามือเหล่านี้จะเป็นท่ามือที่ไม่มีการเคลื่อนไหวของมือ (Static Posture) ซึ่งผลการเปรียบเทียบวิธีการรู้จำทั้ง 3 วิธี สรุปได้ว่า วิธีการที่ให้ค่าความถูกต้องสูงสุด คือ Bayesian Classification ซึ่งค่าความถูกต้องที่ได้ประมาณ 84.66% โดยมีส่วนเบี่ยงเบนมาตรฐานของค่าความถูกต้องเท่ากับ 2.94

2.1.2 งานวิจัยของ Worawit Werapan และ Nopporn Chotikakamthorn [2]

ได้ทำการศึกษาท่ามือภาษาไทย ซึ่งท่ามือภาษาไทยนั้นแบ่งออกเป็น 2 ส่วน คือ ส่วนที่เป็นท่ามือนิ่ง (Static Gesture or Posture) และส่วนที่เป็นท่ามือที่มีการเคลื่อนไหวของมือ (Dynamic Gesture) ซึ่งปัญหาที่พบในการรู้จำท่ามือที่เป็นท่าต่อเนื่อง (Continuous Gesture) ก็คือ จะมีวิธีการอย่างไรที่จะสามารถแยกการเคลื่อนไหวของท่ามือ ซึ่งจะประกอบไปด้วยการเคลื่อนไหวของมือที่มีความหมาย (Meaningful Dynamic Gesture) และการเคลื่อนไหวของมือที่ไม่มีความหมายในตัวเอง (Transitional Gesture) นอกจากนั้นยังมีปัญหาเรื่องความเร็วในการทำท่ามือ, ตำแหน่งของท่ามือ และทิศทางการหันฝ่ามือ ซึ่งแต่ละคนจะมีการทำที่ไม่เหมือนกันหรือไม่เท่ากัน โดยในงานวิจัยนี้ยังได้นำเสนอวิธีการในการแก้ไขปัญหาดังกล่าว โดยยึดหลักที่ว่าท่ามือภาษาไทยและท่ามือภาษาอื่น การเคลื่อนไหวของมือที่มีความหมายนั้น จะเป็นการเคลื่อนที่แบบซ้ำ ๆ (Periodic) ดังนั้นการแยกท่ามือที่มีการเคลื่อนไหวของมือ (Dynamic Gesture) ออกจากการเคลื่อนที่ของมือที่ไม่มีความหมายได้ โดยใช้ Fourier Analysis กับข้อมูลในแต่ละช่วงเวลา ซึ่งข้อมูลที่ได้โดยการใช้ถุงมืออิเล็กทรอนิกส์ที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียกว่า CyberGlove ในกรณีของท่ามือที่มีการเคลื่อนไหวของมือ (Dynamic Gesture) การประมาณค่าช่วงสัญญาณของท่ามือที่ได้จากการวิเคราะห์สามารถนำไปทำ Time Normalization ดังนั้น จึงสามารถลดความผิดพลาดเนื่องจากการทำท่ามือที่มีความเร็วไม่เท่ากันของแต่ละคนได้ การทดลองจะใช้ข้อมูลที่เป็นท่ามือนิ่งและท่ามือที่มีการเคลื่อนไหวอย่างละ 100 ท่า และใช้วิธีการรู้จำที่เรียกว่า Bayesian Estimator ซึ่งผลการทดลองที่ได้ปรากฏว่ามีความถูกต้องในการรู้จำถึง 100% ซึ่งระบบการแปลท่ามือภาษาไทยของ Worawit Werapan และ Nopporn Chotikakamthorn แสดงดังรูปที่ 2.1

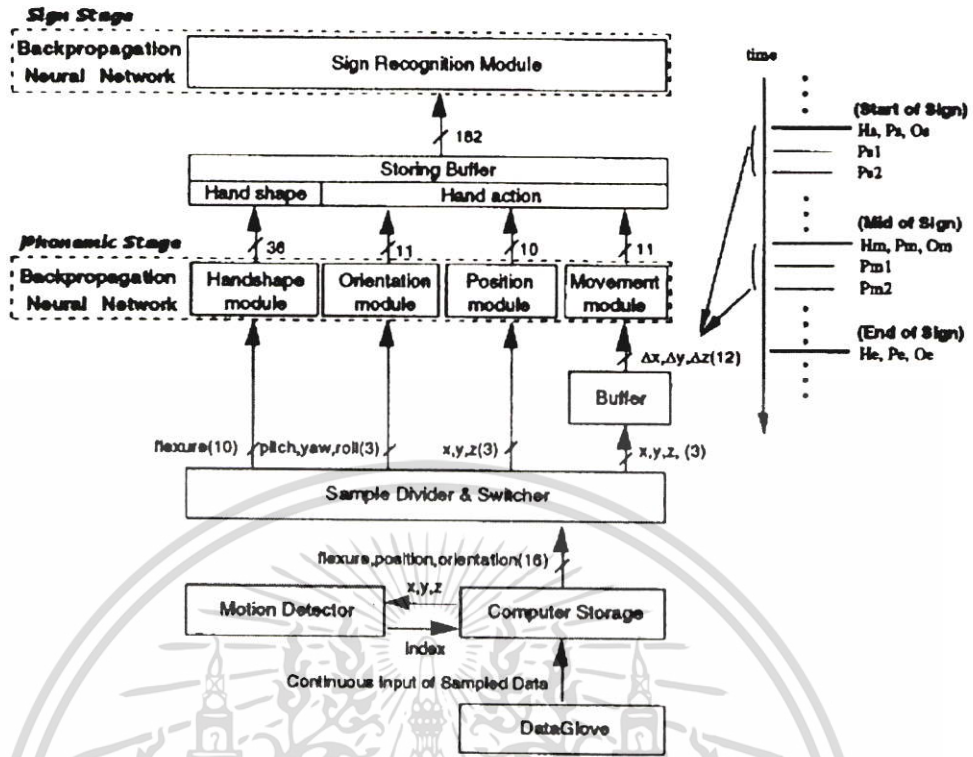


รูปที่ 2.1 Thai Sign Language Translation System

2.1.3 งานวิจัยของ Manjula B. Waldron และ Soowon Kim [3]

ได้ทำการวิจัยเกี่ยวกับระบบรู้จำท่ามืออเมริกัน (American Sign Language) ดังแสดงในรูปที่ 2.2 ในงานวิจัยนี้ได้ออกแบบระบบที่ใช้ในการรู้จำท่ามืออเมริกัน ซึ่งระบบนี้ประกอบไปด้วยการรู้จำ 2 ส่วน คือ ส่วนที่ 1 เป็นการรู้จำส่วนประกอบย่อยของท่ามือ (Phoneme) ซึ่งประกอบไปด้วย Hand shape, Orientation, Position และ Movement ของท่ามือ โดยใช้โครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron ที่ใช้ Back-Propagation algorithm ในการเรียนรู้ ทั้งหมด 4 โครงข่ายเพื่อการรู้จำ Phoneme ทั้ง 4 ตัว และในส่วนที่ 2 ของระบบนี้จะเป็นการรู้จำความหมายของท่ามือ โดยการรับข้อมูลจากโครงข่ายประสาทเทียมทั้ง 4 โครงข่ายในส่วนที่ 1 ในส่วนนี้จะใช้โครงข่ายประสาทเทียม 2 แบบเพื่อเปรียบเทียบ คือ โครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron (MLP) และโครงข่ายประสาทเทียมแบบ Self-Organize Map (SOM) ซึ่งข้อมูลที่ใช้ได้จากถุงมืออิเล็กทรอนิกส์ที่เรียกว่า DataGlove และ Polhemus sensor ซึ่งข้อมูลที่เก็บประกอบด้วย Phoneme ของ Hand shape ทั้งหมด 36 ท่า Position ทั้งหมด 10 ตำแหน่ง Orientation 10 orientation และ Movement 11 movement โดยแบ่งการเคลื่อนไหวของภาษามือในแต่ละคำออกเป็น 3 ส่วนคือ Start of Sign, Mid of Sign และ End of Sign ผลการทดลองในส่วนที่ 1 สรุปได้คือ การรู้จำ Phoneme ของ Hand shape ค่าความถูกต้องในการรู้จำ 93%, การรู้จำ Phoneme ของ Location ค่าความถูกต้องในการรู้จำ 100%, การรู้จำ Phoneme ของ Hand Orientation ค่าความถูกต้องในการรู้จำ 82% และการรู้จำ Phoneme ของ Hand Movement ค่าความถูกต้องในการรู้จำ 82% และในส่วนที่ 2 ของระบบ ในการรู้จำความหมายของท่ามือ ที่ใช้โครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron (MLP) จะให้ความถูกต้องในการรู้จำ 86.2% ส่วนการใช้โครงข่ายประสาทเทียมแบบ Self-Organize Map (SOM) จะให้ความถูกต้องในการรู้จำ 78%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



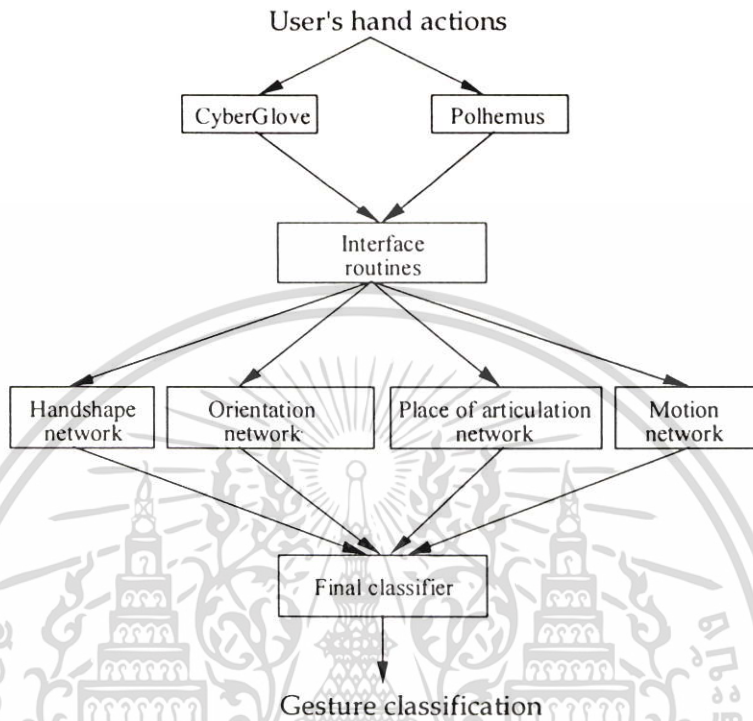
รูปที่ 2.2 ระบบแปลภาษาของ Manjula B. Waldron และ Soowon Kim

2.1.4 งานวิจัยของ S. Sidney Fels และ Geoffrey E. Hinton [4]

ได้ทำการพัฒนาระบบที่เรียกว่า “Glove-Talk” ซึ่งเป็นระบบที่ใช้ในการแปลงข้อมูลที่ได้จากถุงมืออิเล็กทรอนิกส์ ที่เรียกว่า VPL Data-Glove ให้ออกมาเป็นเสียง (Speech) โดยใช้โครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron (MLP) ซึ่งใช้ Back-propagation algorithm ในการเรียนรู้ของโครงข่าย ซึ่งใช้โครงข่ายทั้งหมด 5 โครงข่าย โดยที่โครงข่าย 4 ตัว จะใช้ในการรู้จำข้อมูลที่เป็นองค์ประกอบย่อยของภาษามือ ซึ่งข้อมูลได้จากถุงมืออิเล็กทรอนิกส์ เรียกว่า VPL Data-Glove ประกอบด้วย ท่ามือ (Hand shape), ทิศทางการเคลื่อนที่ของมือ (Orientation), ความเร็วในการเคลื่อนที่ของมือและช่วงเวลาในการทำท่ามือ ซึ่งในระบบนี้ จะมีการตรวจสอบการเก็บข้อมูล กล่าวคือ มีการตรวจสอบว่า ณ ช่วงเวลาใดควรจะมีการเก็บข้อมูลจากถุงมือ เพื่อหลีกเลี่ยง delay ในการแปลงเสียงออกมาเป็นคำ โดยใช้โครงข่ายประสาทเทียมอีก 1 โครงข่ายเป็นตัวตรวจสอบ โดยท่ามือที่ใช้ในการทดลองมีทั้งหมด 203 ท่า ผลการทดลองปรากฏว่า ความถูกต้องในการแปลงเสียงออกมาเป็นคำมีความผิดพลาดน้อยกว่า 1% และความผิดพลาดที่ระบบไม่สามารถแปลงเสียงได้มีน้อยกว่า 5%

2.1.5 งานวิจัยของ Peter Vamplew [5]

ได้ทำการพัฒนาระบบที่เรียกว่า “SLARTI” ดังแสดงในรูปที่ 2.2 ซึ่งเป็นระบบที่ใช้ในการรู้จำท่ามือออสเตรเลีย (Australian sign language หรือ Auslan) ในภาษามือออสเตรเลีย



รูปที่ 2.3 ระบบ SLARTI

ซึ่งภายในระบบจะแบ่งเป็น 2 ส่วน คือ ส่วนที่ 1 จะใช้โครงข่ายประสาทเทียม 4 โครงข่าย เพื่อใช้ในการรู้จำองค์ประกอบย่อยของภาษามือออสเตรเลีย คือ ท่ามือ (Hand shape), การหมุนหรือการหักของฝ่ามือ (Orientation), ตำแหน่งของมือ (Location) และการเคลื่อนที่ของมือ (Movement) ซึ่งระบบ SLARTI จะใช้โครงข่ายประสาทเทียมแบบ Mult-Layer Perceptron (MLP) neural network ที่ใช้ Back-propagation algorithm อุปกรณ์ที่ใช้ในการเก็บข้อมูล คือถุงมืออิเล็กทรอนิกส์ที่เรียกว่า CyberGlove และ Polhemus IsoTrak ในการทดลองจะใช้ผู้ทดลองทั้งหมด 10 คน ข้อมูลที่ได้จะแบ่งออกเป็น 2 ส่วนคือ ข้อมูลของ 7 คนแรก จะนำไปใช้ทั้งในการเรียนรู้และทดสอบโครงข่าย เรียกว่า Registered Data และข้อมูลของ 3 คนที่เหลือจะใช้เฉพาะทดสอบโครงข่ายเท่านั้น เรียกว่า Unregistered Data ผลการทดลองสามารถสรุปได้ดังนี้ ในการรู้จำท่ามือ Registered Data จะให้ค่าความถูกต้องประมาณ 97.4% Unregistered Data จะให้ค่าความถูกต้องประมาณ 89.5%, ในการรู้จำการหมุนหรือการหักของฝ่ามือ Registered Data จะให้ค่าความถูกต้องประมาณ 91.6% Unregistered Data จะให้ค่าความถูกต้องประมาณ 89.2%, ในการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รู้จำตำแหน่งของมือ Registered Data จะให้ค่าความถูกต้องประมาณ 76.4% Unregistered Data จะให้ค่าความถูกต้องประมาณ 69.0% และในการรู้จำการเคลื่อนไหวของมือ Registered Data จะให้ค่าความถูกต้องประมาณ 92.3% Unregistered Data จะให้ค่าความถูกต้องประมาณ 76.9% ในส่วนที่สองของระบบนี้ จะเปรียบเทียบวิธีการในการรู้จำ 2 วิธี คือ วิธีการแบบ Nearest-Neighbour Lookup Algorithm และวิธีการแบบ C4.5 Decision Tree ในการรู้จำ ซึ่งผลปรากฏว่า วิธีการแบบ nearest-neighbour lookup algorithm จะให้ค่าความถูกต้องที่ดีกว่า คือ Registered Data จะให้ค่าความถูกต้องประมาณ 94% Unregistered Data จะให้ค่าความถูกต้องประมาณ 85%



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

หลักการรู้จำรูปแบบ

ปัจจุบันมีวิธีการที่ใช้ในการรู้จำรูปแบบมากมายหลายวิธี ในแต่ละวิธีก็จะมีข้อดีและข้อด้อยที่แตกต่างกันไป ซึ่งในงานวิจัยขั้นนี้ จะทำการศึกษาเปรียบเทียบวิธีการที่เกี่ยวข้องกับการรู้จำ 2 วิธี คือ วิธีการที่อ้างอิงทางด้านสถิติ เช่น Bayesian Estimator และวิธีการแบบโครงข่ายประสาทเทียม (Neural Network) โดยวิธีการแบบโครงข่ายประสาทเทียมจะทำการศึกษาทั้งหมด 2 รูปแบบ คือ Multi-Layer Perceptron (MLP) Neural Network และ Radial Basis Function (RBF) Neural Network

3.1 Bayesian Estimator

วิธีการแบบ Bayesian Estimator [6-7] เป็นวิธีการที่อ้างอิงกับสถิติ กล่าวคือ สมการที่นิยามใช้ในวิธีการแบบ Bayesian Estimator จะเป็นสมการของการกระจายแบบปกติ (Normal Distribution) ดังสมการที่ 3.1

$$p(x|\omega_i) = \frac{1}{\sqrt{(2\pi)^n |C_i|}} \exp\left[-\frac{1}{2} (x - \mu_i)^t C_i^{-1} (x - \mu_i)\right] \quad (3.1)$$

โดยที่

$p(x|\omega_i)$ คือ ความน่าจะเป็นของข้อมูล x เมื่อกำหนดข้อมูลของคลาส i^h ให้ (class-conditional probability density function)

x คือ เวกเตอร์ข้อมูล

n คือ มิติของข้อมูล (dimension)

μ_i คือ ค่าเฉลี่ยของคลาส i^h

C_i^{-1} คือ เมทริกซ์ส่วนกลับของค่าความแปรปรวนร่วม (Inverse Covariance) ของคลาส i^h

C_i คือ ค่า determinant ของค่าความแปรปรวนร่วม คลาส i^h

จากสมการที่ 3.1 สามารถหาค่า Normal Probability Density Function ของคลาส i^h เมื่อกำหนดค่าเวกเตอร์ข้อมูล x ได้ ดังสมการที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$p(\omega_i | \mathbf{x}) = \frac{p(\mathbf{x} | \omega_i)p(\omega_i)}{p(\mathbf{x})} \quad (3.2)$$

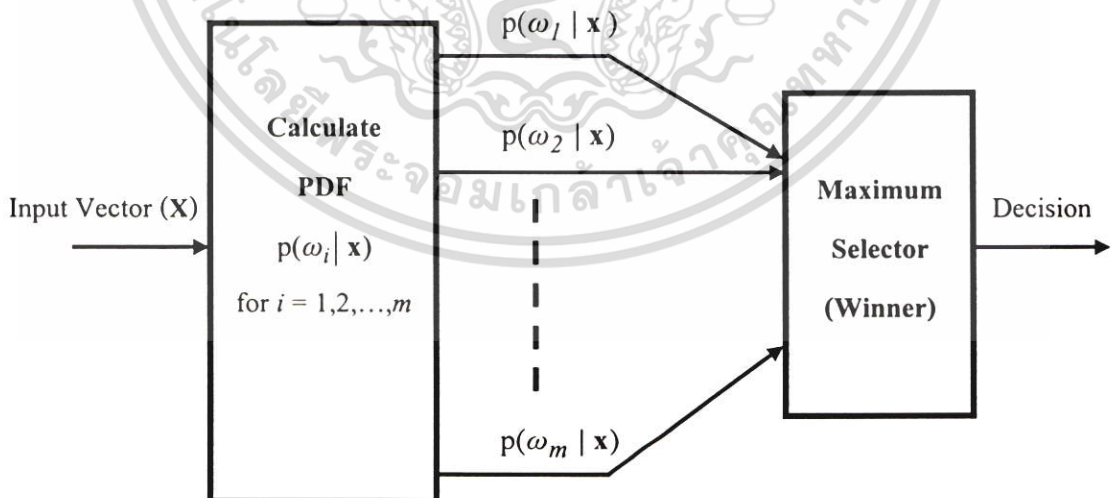
โดยที่

- $p(\mathbf{x})$ คือ ความน่าจะเป็นที่ \mathbf{x} จะเกิดขึ้น (probability density function)
 $p(\omega_i)$ คือ ความน่าจะเป็นของคลาส (class) i^{th}
 $p(\omega_i | \mathbf{x})$ คือ ความน่าจะเป็นของคลาส i^{th} โดยกำหนด \mathbf{x} ให้
 (posterior probability)

สมการที่ 3.2 เป็นสมการที่มาจากทฤษฎีของเบย์ (Baye's Theory) ซึ่งการรู้จำข้อมูลโดยใช้ทฤษฎีของเบย์ สามารถทำได้โดยการเปรียบเทียบค่า $p(\omega_i | \mathbf{x})$ ของคลาสทั้งหมด ซึ่งข้อมูลจะถูกกำหนดให้อยู่ในคลาสที่มีค่า $p(\omega_i | \mathbf{x})$ สูงที่สุด ซึ่งเรียกกฎนี้ว่า Baye's Optimum หรือ Maximum-Likelihood Decision Rule [6-7] ดังสมการที่ 3.3

$$\mathbf{x} \in \omega_i \text{ if } \frac{p(\mathbf{x} | \omega_i)p(\omega_i)}{p(\mathbf{x})} \geq \frac{p(\mathbf{x} | \omega_j)p(\omega_j)}{p(\mathbf{x})} \text{ for all } j \neq i \quad (3.3)$$

จากสมการที่ 3.3 สามารถแสดงขั้นตอนการทำงานของวิธีการแบบ Bayesian Estimator ได้ ดังรูปที่ 3.1



รูปที่ 3.1 ขั้นตอนการทำงานของวิธีการแบบ Bayesian Estimator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในทางปฏิบัติแล้ว จะกำหนด discriminant function, $d_i(\mathbf{x})$ สำหรับแต่ละคลาส เพื่อใช้ในการเปรียบเทียบ ดังสมการที่ 3.4

$$d_i(\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)p(\omega_i)}{p(\mathbf{x})} \quad (3.4)$$

โดยทั่วไปจะพยายามที่จะลดความซับซ้อนของ discriminant function เพื่อลดการคำนวณของฟังก์ชัน โดยที่ไม่มีผลต่อประสิทธิภาพของการรู้จำ จากสมการที่ 3.4 สามารถตัดเทอม $p(\mathbf{x})$ ได้ เนื่องจากว่า มีค่าเท่ากัน ซึ่งจะสนใจเฉพาะเทอมที่มีความเกี่ยวข้องกับคลาสเท่านั้น สมการที่ได้จะเป็นดังสมการที่ 3.5

$$d_i(\mathbf{x}) = p(\mathbf{x}|\omega_i)p(\omega_i) \quad (3.5)$$

ในกรณีที่ค่าความน่าจะเป็นของทุก ๆ คลาส มีค่าเท่ากัน ก็สามารถลดการคำนวณของสมการที่ 3.5 ได้ โดยการตัดเทอม $p(\omega_i)$ ดังนั้น สมการที่ได้จะเป็นดังสมการที่ 3.6

$$d_i(\mathbf{x}) = p(\mathbf{x}|\omega_i) \quad (3.6)$$

จากสมการที่ 3.1 และสมการที่ 3.6 สามารถเขียนสมการใหม่ให้อยู่ในรูปของ discriminant function ได้ ดังสมการที่ 3.7

$$d_i(\mathbf{x}) = \frac{1}{\sqrt{(2\pi)^n |C_i|}} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)' C_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)\right] \quad (3.7)$$

จากสมการที่ 3.7 สามารถลดความซับซ้อนในการคำนวณของฟังก์ชัน Exponential โดยใช้ลอการิทึมธรรมชาติ (Natural Logarithmic) สมการที่ได้จะเป็นดังสมการที่ 3.8

$$d_i(\mathbf{x}) = -\frac{1}{2} \ln(2\pi)^n - \frac{1}{2} \ln|C_i| - \frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_i)' C_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i) \quad (3.8)$$

จากสมการที่ 3.8 สามารถที่จะตัดเทอมที่ 1 ออกไปได้ เนื่องจากว่าเป็นค่าคงที่ สมการที่ได้จะเป็นดังสมการที่ 3.9

$$d_i(\mathbf{x}) = -\frac{1}{2} \ln|C_i| - \frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_i)' C_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i) \quad (3.9)$$

จากสมการที่ 3.9 สามารถเขียนให้อยู่ในรูปสมการอย่างง่าย ได้ดังสมการที่ 3.10

$$d_i(\mathbf{x}) = (\mathbf{x} - \boldsymbol{\mu}_i)' C_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i) + \ln|C_i| \quad (3.10)$$

3.2 โครงข่ายประสาทเทียม (Neural Network)

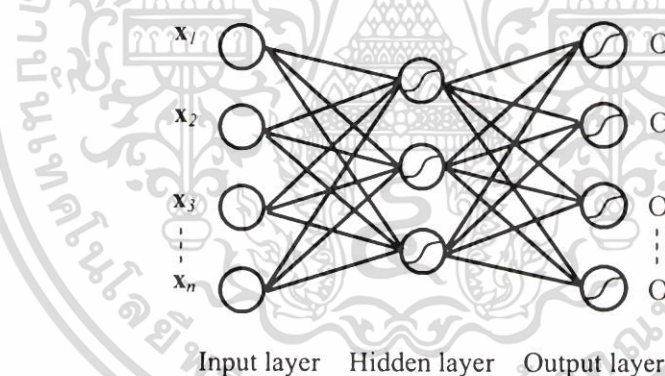
วิธีการแบบโครงข่ายประสาทเทียม มีมากมายหลายแบบ ในงานวิจัยฉบับนี้ จะทำการศึกษาโครงข่ายประสาทเทียมเพียง 2 แบบ คือ

- Multi-Layer Perceptron Neural Network (MLP)
- Radial Basis Function Neural Network (RBF)

3.2.1 Multi-Layer Perceptron Neural Network (MLP) [8-10]

3.2.1.1 โครงสร้างของโครงข่าย

โครงสร้างทั่วไปของโครงข่ายประสาทเทียมแบบ MLP แสดงในรูปที่ 3.2



รูปที่ 3.2 โครงสร้างของโครงข่ายประสาทเทียมแบบ MLP

จากรูปที่ 3.2 โครงข่ายประสาทเทียมแบบ MLP มีโครงสร้าง ดังนี้คือ

- โครงข่ายประกอบด้วยชั้น (Layer) ต่าง ๆ หลายชั้น คือ ชั้นอินพุต (Input Layer) ชั้นซ่อน (Hidden Layer) และชั้นผลลัพธ์ (Output Layer)
- ในแต่ละชั้นที่ติดกัน โหนดทุกโหนดจะเชื่อมต่อกันแบบทั่วถึง (Fully Connect) โดยค่าของ Weight

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.2 การเรียนรู้ของโครงข่าย

โครงข่ายประสาทเทียมแบบ MLP จะใช้ Back propagation (BP) algorithm [8-10] ในการเรียนรู้ ซึ่งอัลกอริทึมจะประกอบด้วย 2 ส่วน คือ ส่วนที่เป็น feed forward และส่วนที่เป็น back-propagation ในสถานะเริ่มต้นของการเรียนรู้ ค่า Weight ของโครงข่ายจะถูกสุ่มขึ้นมา และค่า Learning Rate (η) ในงานวิจัยนี้กำหนดให้มีค่าเท่ากับ 0.01 กำหนดให้ x_i คือเวกเตอร์ข้อมูลมีขนาด n มิติ จะถูกส่งเข้าสู่ชั้นอินพุทของโครงข่าย โดยที่ โหนดที่ i^{th} ของชั้นอินพุทจะถูกเชื่อมไปยังโหนดที่ j^{th} ของชั้นซ่อน ผ่านค่า W_{ji} โดยที่ $i = 1, 2, \dots, n$ และค่า $j = 1, 2, \dots, s$ โหนดที่ j^{th} ของชั้นซ่อนจะถูกเชื่อมไปยังโหนดที่ k^{th} ของชั้นผลลัพธ์ ผ่านค่า Z_{kj} โดยที่ $k = 1, 2, \dots, m$

โดยที่

n = จำนวนโหนดในชั้นอินพุท

s = จำนวนโหนดในชั้นซ่อน

m = จำนวนโหนดในชั้นผลลัพธ์

ผลลัพธ์ของชั้นซ่อน คำนวณได้โดยการนำค่าเวกเตอร์ข้อมูลคูณกับค่า Weight (W) ที่เกี่ยวข้อง แล้วนำค่าผลลัพธ์ที่ได้มาผ่านฟังก์ชันเพื่อหาค่าผลลัพธ์ ดังสมการที่ 3.10

$$h_j = f(\text{net}_j) \quad (3.10)$$

$$\text{net}_j = \sum_{i=1}^n W_{ji} x_i \quad (3.11)$$

โดยที่ $f(\bullet)$ คือ Activate Function หรือ Transfer function โดยทั่วไปจะเป็น Sigmoid Function ดังสมการที่ 3.12

$$f(x) = \frac{1}{1 + \exp(-x)} \quad (3.12)$$

หลังจากได้ผลลัพธ์ที่ชั้นซ่อนแล้ว ผลลัพธ์ที่ชั้นซ่อนก็จะกลายเป็นอินพุทถูกส่งต่อไปยังชั้นผลลัพธ์ ซึ่งผลลัพธ์คำนวณได้โดยการนำค่าอินพุทกับค่า Weight (Z) มา dot product หรือ inner product กัน ดังสมการที่ 3.13 – 3.14

$$o_k = f(\text{net}_k) \quad (3.13)$$

$$\text{net}_k = \sum_{j=1}^s Z_{kj} h_j \quad (3.14)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เวกเตอร์ของผลลัพธ์ (\mathbf{o}_k), โดยที่ $\mathbf{o}_k = [o_1, o_2, \dots, o_m]$ ที่ได้จากระบวนการในส่วน
ของ feed forward จะถูกนำไปเปรียบเทียบกับค่าเวกเตอร์ของผลลัพธ์ที่ต้องการ (Desire Output)
(\mathbf{t}_k), โดยที่ $\mathbf{t}_k = [t_1, t_2, \dots, t_m]$ ซึ่งค่าความผิดพลาด (Error) หรือ Cost Function ที่ใช้ในงานวิจัยนี้
คือ square error function ดังสมการที่ 3.15

$$e = \frac{1}{2} \sum_{k=1}^m (\mathbf{t}_k - \mathbf{o}_k)^2 \quad (3.15)$$

หลังจากที่คำนวณในส่วนของ feed forward เรียบร้อยแล้ว ก็มาถึงกระบวนการในส่วน
ของ back-propagation ซึ่งจะใช้ค่าความผิดพลาด ($\mathbf{t}_k - \mathbf{o}_k$) เพื่อจะนำไปใช้ในการปรับค่า Weight
โดยสมการที่ 3.16 – 3.19

$$\mathbf{z}_{(t+1)} = \eta \delta_k \mathbf{h}_j + \mathbf{z}_{(t)} \quad (3.16)$$

โดยที่

$$\delta_k = \mathbf{o}_k (1 - \mathbf{o}_k) (\mathbf{t}_k - \mathbf{o}_k) \quad (3.17)$$

$$\mathbf{w}_{(t+1)} = \eta \beta_j \mathbf{x}_i + \mathbf{w}_{(t)} \quad (3.18)$$

โดยที่

$$\beta_j = \mathbf{h}_j (1 - \mathbf{h}_j) \sum_k \delta_k \mathbf{w}_{kj} \quad (3.19)$$

หลังจากการปรับค่า Weight เรียบร้อยแล้ว ก็มาถึงขั้นตอนของการตรวจสอบเงื่อนไข
ว่ากระบวนการเรียนรู้ของโครงข่ายควรจะหยุดหรือจะเรียนรู้ต่อไป ซึ่งเงื่อนไขที่ใช้ก็มีหลายเงื่อนไข
เช่น

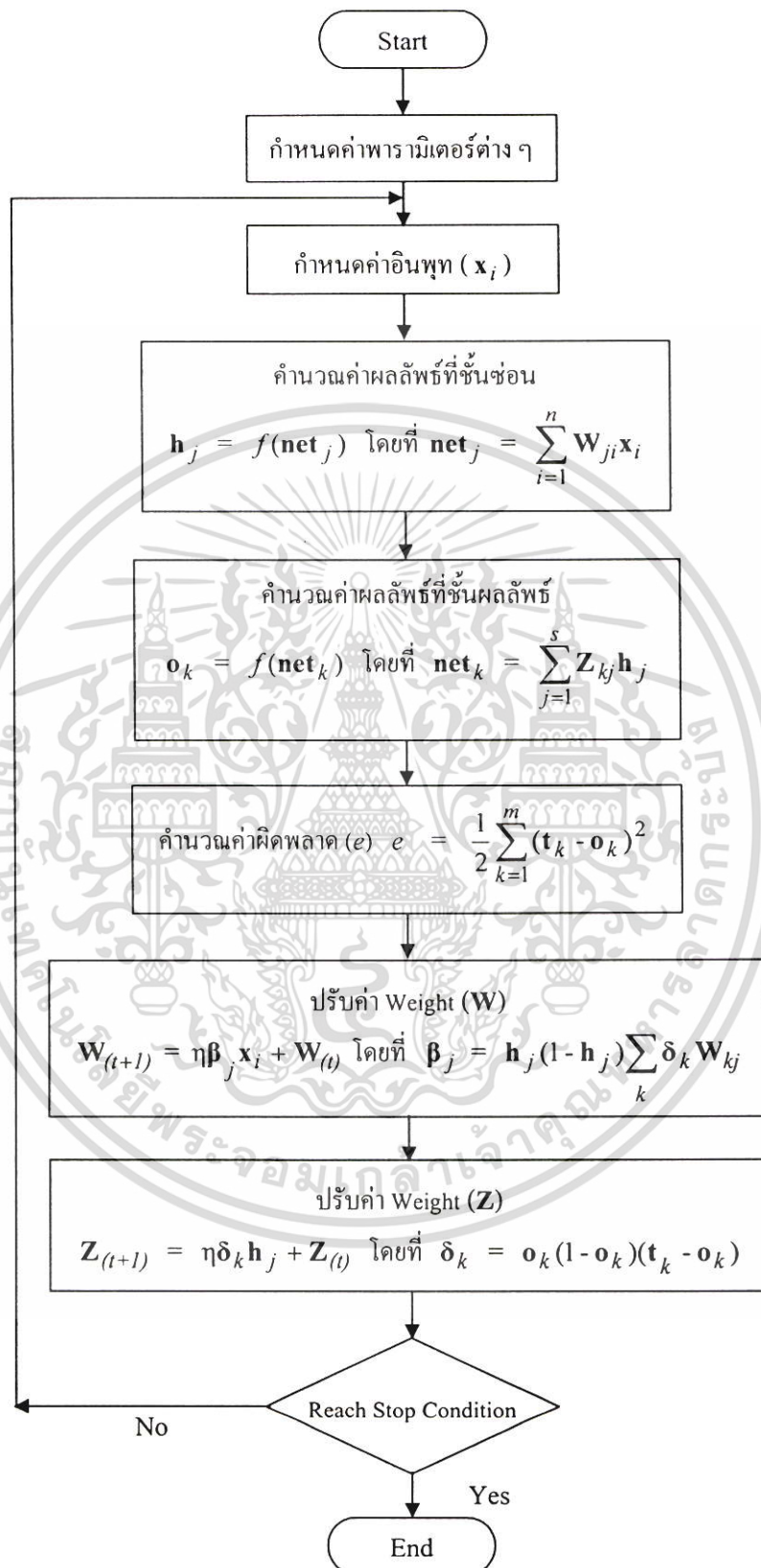
- การใช้ค่าความผิดพลาด (Error) หรือ Cost Function ดังสมการที่ 3.15
เปรียบเทียบกับค่าความผิดพลาดที่ต้องการ (Desire Error) หรือค่า Threshold
กล่าวคือ ถ้าหากว่าค่าความผิดพลาด (Error) หรือ Cost Function ต่ำกว่าหรือ
เท่ากับค่าความผิดพลาดที่ต้องการก็จะหยุดการเรียนรู้
- การกำหนดจำนวนรอบ (Epoch) ของการเรียนรู้ที่ต้องการ กล่าวคือ จะมีการ
กำหนดจำนวนรอบที่ต้องการเรียนรู้ของโครงข่าย เมื่อโครงข่ายเรียนรู้จนครบ
ตามจำนวนรอบที่กำหนด ก็จะหยุดการเรียนรู้

ในงานวิจัยนี้ เงื่อนไขที่ใช้คือ การกำหนดจำนวนรอบ ซึ่งจากผลการทดลอง พบว่า

จำนวนรอบที่ทำให้ค่าความถูกต้องในการรู้จำดีที่สุด คือ 1,500 รอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งวิธีการเรียนรู้แบบ Back-propagation algorithm สามารถสรุปได้ ดังรูปที่ 3.3



รูปที่ 3.3 ไดอะแกรมแสดงขั้นตอนการเรียนรู้ของโครงข่ายประสาทเทียมแบบ MLP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.3 การรู้จำของโครงข่าย

หลังจากโครงข่ายประสาทเทียมได้รับการเรียนรู้แล้ว สามารถที่จะนำมาใช้ในการรู้จำข้อมูลได้ ซึ่งขั้นตอนในการรู้จำของโครงข่าย สามารถสรุปได้ ดังนี้

0. กำหนดให้

x_i คือ เวกเตอร์ข้อมูลที่ต้องการรู้จำ

W_{ji} คือ ค่า Weight ที่เชื่อมจากโหนด i^{th} ของชั้นอินพุตไปยังโหนด j^{th} ของชั้นซ่อน

Z_{kj} คือ ค่า Weight ที่เชื่อมจากโหนด j^{th} ของชั้นซ่อนไปยังโหนด k^{th} ของชั้นผลลัพธ์

1. เมื่อเวกเตอร์ข้อมูล (x_i) ถูกส่งเข้าสู่โครงข่ายโดยผ่านชั้นอินพุต (Input Layer) และถูกส่งต่อไปยังชั้นซ่อน เพื่อคำนวณหาค่าผลลัพธ์โดยผ่านค่า Weight (W_{ji}) ซึ่งผลลัพธ์ของชั้นซ่อน สามารถคำนวณโดยสมการที่ 3.20 – 3.21

$$h_j = f(\text{net}_j) \quad (3.20)$$

$$\text{net}_j = \sum_{i=1}^n W_{ji} x_i \quad (3.21)$$

2. คำนวณหาค่าผลลัพธ์ที่ชั้นผลลัพธ์ (Output Layer) ซึ่งผลลัพธ์ของชั้นผลลัพธ์ สามารถคำนวณ โดยสมการที่ 3.22 – 3.23

$$o_k = f(\text{net}_k) \quad (3.22)$$

$$\text{net}_k = \sum_{j=1}^s Z_{kj} h_j \quad (3.23)$$

3. เวกเตอร์ผลลัพธ์ (o_k) ที่ได้จากขั้นตอนที่ 2 นำไปหาตำแหน่งที่ค่าในเวกเตอร์ผลลัพธ์มีค่ามากที่สุด เพื่อหาว่าเวกเตอร์ข้อมูลควรจะเป็นสมาชิกในคลาสใด เช่น สมมติว่าตำแหน่งที่ k^{th} ของเวกเตอร์ผลลัพธ์มีค่ามากที่สุด หมายความว่า เวกเตอร์ข้อมูลควรจะเป็นสมาชิกในคลาส k^{th} ดังสมการที่ 3.24

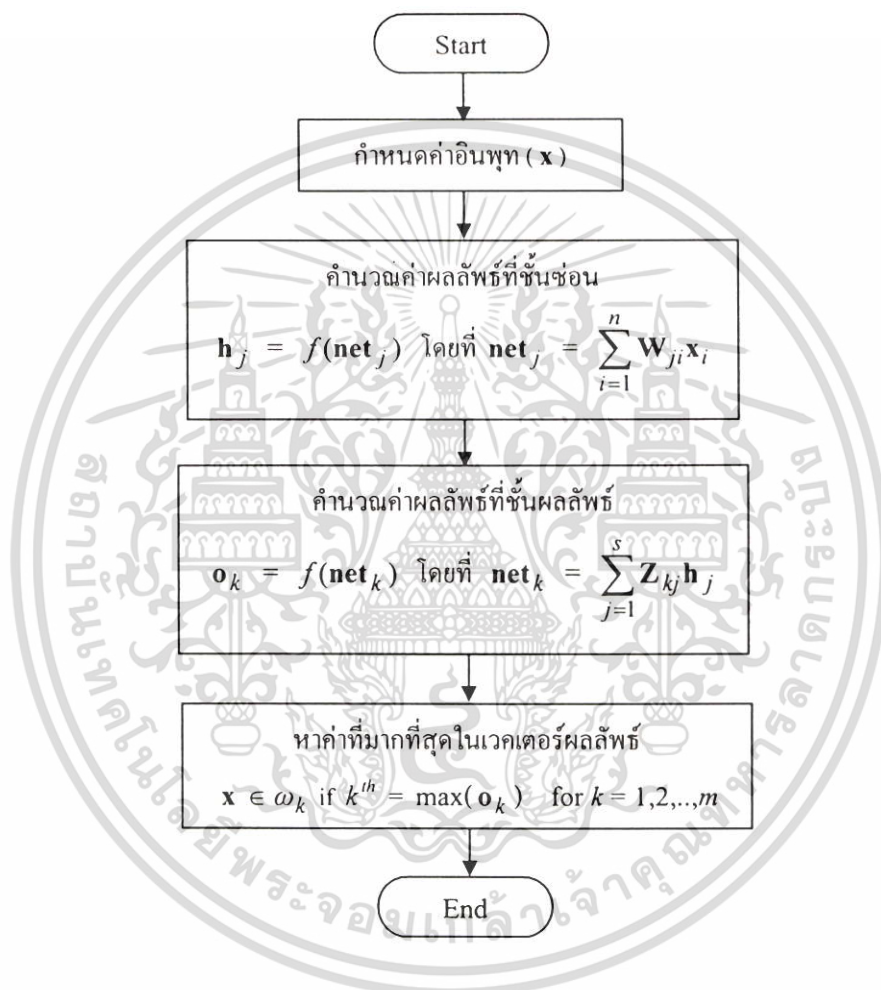
$$x \in \omega_k \text{ if } k^{th} = \max(o_k) \text{ for } k = 1, 2, \dots, m \quad (3.24)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่

- m คือ มิติของเวกเตอร์ผลลัพธ์ (จำนวน โหนดในชั้นผลลัพธ์)
- k^{th} คือ ตำแหน่งที่มีค่ามากที่สุด ในเวกเตอร์ผลลัพธ์
- ω_k คือ คลาสที่เวกเตอร์ข้อมูลเป็นสมาชิก (Winner Class)

ซึ่งสามารถสรุปขั้นตอนการรู้จำของโครงข่ายได้ ดังรูปที่ 3.4



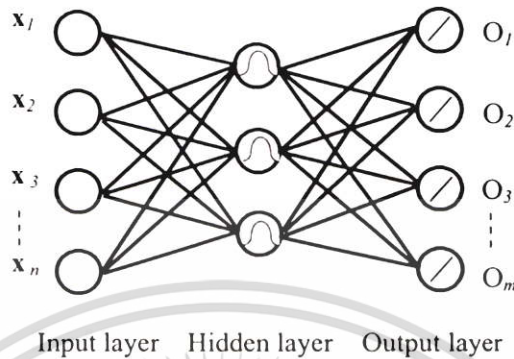
รูปที่ 3.4 ไคอะแกรมแสดงขั้นตอนการรู้จำของโครงข่ายประสาทเทียมแบบ MLP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 Radial Basis Function Neural Network (RBF) [8-11]

3.2.2.1 โครงสร้างของโครงข่าย

โครงสร้างทั่วไปของโครงข่ายประสาทเทียมแบบ RBF แสดงดังรูปที่ 3.5



รูปที่ 3.5 โครงสร้างของโครงข่ายประสาทเทียมแบบ Radial Basis Function Network

โครงสร้างของโครงข่ายประสาทเทียมแบบ RBF จะมีลักษณะที่คล้ายกับโครงสร้างของโครงข่ายประสาทเทียมแบบ MLP ดังนี้

- โครงข่ายประกอบด้วยชั้น (Layer) ต่าง ๆ คือ ชั้นอินพุท (Input Layer) ชั้นซ่อน (Hidden Layer) และชั้นผลลัพธ์ (Output Layer)
- ในแต่ละชั้นที่ติดกัน โหนดทุกโหนดจะเชื่อมต่อกันแบบทั่วถึง (Fully Connect) โดยค่าของ Weight

โครงสร้างของโครงข่ายประสาทเทียมแบบ RBF จะมีลักษณะที่แตกต่างจากโครงสร้างของโครงข่ายประสาทเทียมแบบ MLP ดังนี้

- Activate Function โครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron ที่ชั้นผลลัพธ์และชั้นซ่อนจะเป็น non-linear function โดยทั่วไปจะเป็น sigmoid function ในขณะที่ Radial Basis Function Network ที่ชั้นผลลัพธ์จะเป็น linear function ส่วนที่ชั้นซ่อนจะเป็น non-linear radial basis function

- การคำนวณหาค่า net input ของ activate function ในชั้นซ่อนของโครงข่ายประสาทเทียมแบบ Radial Basis Function จะใช้ Euclidean norm ระหว่างเวกเตอร์ข้อมูลกับค่า Weight ของแต่ละโหนด ส่วนของโครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron จะใช้ dot product หรือ inner product ระหว่างค่าเวกเตอร์ข้อมูลกับค่า Weight ของแต่ละโหนดในชั้นผลลัพธ์

- จำนวนชั้นซ่อน ซึ่งโครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron จะมีจำนวนชั้นซ่อน 1 หรือมากกว่า 1 ชั้นก็ได้ แต่โครงข่ายประสาทเทียมแบบ Radial Basis Function โดยทั่วไปจะมีเพียง 1 ชั้นเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2.2 การเรียนรู้ของโครงข่าย

Radial Basis Function Network มีวิธีการในการเรียนรู้มากมายหลายแบบ ซึ่งในงานวิจัยนี้จะใช้การเรียนรู้โดยวิธี Gradient Descent ซึ่งเป็นการเรียนรู้แบบ Supervise Training ที่มีลักษณะคล้ายกับ back-propagation algorithm ที่ใช้ในการเรียนรู้ของ Multi-Layer Perceptron กล่าวคือ จะประกอบด้วย 2 ส่วน คือ ส่วนของ forward และ backward ซึ่งสรุปได้ดังนี้

ค่าพารามิเตอร์ที่ใช้สำหรับการเรียนรู้ของโครงข่ายประสาทเทียมแบบ RBF มีดังนี้

- η คือ ค่า Learning Rate
- Z_{kj} คือ ค่า Weight ที่เชื่อมระหว่างโหนด j^{th} ของชั้นซ่อนไปยังโหนด k^{th} ของชั้นผลลัพธ์
- μ_j คือ ค่า Center ของโหนด j^{th} ในชั้นซ่อน ซึ่งจะมีขนาดเท่ากับขนาดของเวกเตอร์ข้อมูล (n มิติ)
- σ_j^2 คือ ค่าความแปรปรวน หรือค่า Width ของ Center โหนด j^{th} ในชั้นซ่อน หรือจะใช้เป็นค่าของส่วนเบี่ยงเบนมาตรฐาน σ_j ก็ได้

ในสถานะเริ่มต้นของการเรียนรู้ ค่าพารามิเตอร์ต่าง ๆ ในงานวิจัยนี้จะกำหนด ดังนี้ คือ

- η จะกำหนดให้มีค่าเท่ากับ 0.01
- Z_{kj} จะกำหนดโดยการสุ่มค่า
- μ_j จะกำหนดโดยการสุ่มเลือกจากข้อมูลที่นำมาใช้ในการเรียนรู้
- σ_j^2 จะกำหนดค่าโดยการหารระยะทางของ Center ระหว่างโหนด j^{th} กับโหนดที่ใกล้กับโหนด j^{th} มากที่สุด ดังสมการที่ 3.25

$$\sigma_j = \sqrt{\frac{1}{p} \sum_{i=1}^p (\mu_j - \mu_{jp})^2} \quad (3.25)$$

โดยที่

p คือ จำนวนของ Center ที่ใกล้กับโหนด j^{th} มากที่สุด ซึ่งในงานวิจัยนี้จะกำหนดให้มีค่าเท่ากับ 2

กำหนดให้ x คือ เวกเตอร์ข้อมูลมีขนาด n มิติ โดยโหนดที่ i^{th} ของชั้นอินพุตจะถูกเชื่อมไปยังโหนดที่ j^{th} ของชั้นซ่อน ผ่านค่า μ_j โดยที่ $i = 1, 2, \dots, n$ และค่า $j = 1, 2, \dots, s$ และโหนดที่ j^{th} ของชั้นซ่อนจะถูกเชื่อมไปยังโหนดที่ k^{th} ของชั้นผลลัพธ์ ผ่านค่า Weight (Z_{kj}) โดยที่ $k = 1, 2, \dots, m$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่

n = จำนวนโหนดในชั้นอินพุต

s = จำนวนโหนดในชั้นซ่อน

m = จำนวนโหนดในชั้นผลลัพธ์

ผลลัพธ์ของชั้นซ่อน คำนวณได้โดยใช้ Radial Basis Function ดังสมการที่ 3.26

$$\mathbf{h}_j = \exp\left(-\frac{\|\mathbf{x} - \boldsymbol{\mu}_j\|^2}{\sigma_j^2}\right) \quad (3.26)$$

ผลลัพธ์ของชั้นซ่อนจะถูกส่งต่อไปยังชั้นผลลัพธ์โดยผ่านค่า Weight (\mathbf{Z}) เพื่อหาค่าผลลัพธ์ของโครงข่าย ซึ่งสามารถคำนวณได้ดังสมการที่ 3.27

$$\mathbf{o}_k = \sum_{j=1}^s \mathbf{Z}_{kj} \mathbf{h}_j \quad (3.27)$$

เวกเตอร์ผลลัพธ์ (\mathbf{o}_k), โดยที่ $\mathbf{o}_k = [o_1, o_2, \dots, o_m]$ ที่ได้จากสมการที่ 3.27 จะถูกนำไปเปรียบเทียบกับค่าเวกเตอร์ผลลัพธ์ที่ต้องการ (Desire Output) (\mathbf{t}_k), เพื่อหาค่าความผิดพลาด โดยที่ $\mathbf{t}_k = [t_1, t_2, \dots, t_m]$ ซึ่งค่าความผิดพลาด (Error) หรือ Cost Function ในงานวิจัยนี้คือ square error function ดังสมการที่ 3.28

$$e = \frac{1}{2} \sum_{k=1}^m (\mathbf{t}_k - \mathbf{o}_k)^2 \quad (3.28)$$

ค่าความผิดพลาดจะนำไปใช้ในการปรับค่า Weight ดังสมการที่ 3.29 – 3.30

$$\mathbf{Z}_{(t+1)} = \eta \mathbf{e}_k \mathbf{h}_j + \mathbf{Z}_{(t)} \quad (3.29)$$

โดยที่

$$\mathbf{e}_k = (\mathbf{t}_k - \mathbf{o}_k) \quad (3.30)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนต่อไป คือ การปรับค่า Center (μ_j) และ σ_j^2 ของโครงข่าย โดยใช้สมการที่ 3.31 และสมการที่ 3.32

$$\mu_j = \mu_j + \eta h_j \frac{\|x - \mu_j\|}{\sigma_j^2} \sum_k e_k Z_{kj} \quad (3.31)$$

$$\sigma_j^2 = \sigma_j^2 + \eta h_j \frac{\|x - \mu_j\|^2}{\sigma_j^3} \sum_k e_k Z_{kj} \quad (3.32)$$

หลังจากการปรับค่า Weight เรียบร้อยแล้ว ก็มาถึงขั้นตอนของการตรวจสอบเงื่อนไขว่ากระบวนการเรียนรู้ของโครงข่ายควรจะหยุดหรือจะเรียนรู้ต่อไป ซึ่งเงื่อนไขที่ใช้ก็มีหลายเงื่อนไข เช่น

- การใช้ค่าความผิดพลาด (Error) หรือ Cost Function ดังสมการที่ 3.28 เปรียบเทียบกับค่าความผิดพลาดที่ต้องการ (Desire Error) หรือค่า Threshold กล่าวคือ ถ้าหากว่าค่าความผิดพลาด (Error) หรือ Cost Function ต่ำกว่าหรือเท่ากับค่าความผิดพลาดที่ต้องการก็จะหยุดการเรียนรู้
- การกำหนดจำนวนรอบ (Epoch) ของการเรียนรู้ที่ต้องการ กล่าวคือ จะมีการกำหนดจำนวนรอบที่ต้องการเรียนรู้ของโครงข่าย เมื่อโครงข่ายเรียนรู้จนครบตามจำนวนรอบที่กำหนด ก็จะหยุดการเรียนรู้

ในงานวิจัยนี้ เงื่อนไขที่ใช้คือ การกำหนดจำนวนรอบ ซึ่งจากผลการทดลอง พบว่าจำนวนรอบที่ทำให้ค่าความถูกต้องในการรู้จำดีที่สุด คือ 500 รอบ

ซึ่งวิธีการเรียนรู้ของ Radial Basis Function Network สามารถสรุปได้ ดังรูปที่ 3.6

3.2.2.3 การรู้จำของโครงข่าย

หลังจากโครงข่ายประสาทเทียมได้รับการเรียนรู้แล้ว สามารถที่จะนำมาใช้ในการรู้จำข้อมูลได้ ซึ่งขั้นตอนในการรู้จำของโครงข่าย สามารถสรุปได้ ดังนี้

0. กำหนดให้

- x คือ เวกเตอร์ข้อมูลที่ต้องการรู้จำ
- Z_{kj} คือ ค่า Weight ที่เชื่อมระหว่างโหนด j^{th} ของชั้นซ่อนไปยังโหนด k^{th} ของชั้นผลลัพธ์
- μ_j คือ ค่า Center ของโหนด j^{th} ในชั้นซ่อน ซึ่งจะมิตขนาดเท่ากับขนาดของเวกเตอร์ข้อมูล (n มิติ)

- σ_j^2 คือ ค่าความแปรปรวน หรือค่า Width ของ Center โหนด j^{th} ในชั้นซ่อน

1. เมื่อเวกเตอร์ข้อมูล (\mathbf{x}) ถูกส่งเข้าสู่โครงข่ายโดยผ่านชั้นอินพุท (Input Layer) และถูกส่งต่อไปยังชั้นซ่อน เพื่อคำนวณหาค่าผลลัพธ์โดยผ่านค่า Center (μ_j) ซึ่งผลลัพธ์ของชั้นซ่อน สามารถคำนวณโดยสมการที่ 3.33

$$\mathbf{h}_j = \exp\left(-\frac{\|\mathbf{x} - \mu_j\|^2}{\sigma_j^2}\right) \quad (3.33)$$

2. คำนวณหาค่าผลลัพธ์ที่ชั้นผลลัพธ์ (Output Layer) ซึ่งผลลัพธ์ของชั้นผลลัพธ์ สามารถคำนวณโดยสมการที่ 3.34 – 3.35

$$\mathbf{o}_k = f(\mathbf{net}_k) \quad (3.34)$$

$$\mathbf{net}_k = \sum_{j=1}^s \mathbf{Z}_{kj} \mathbf{h}_j \quad (3.35)$$

3. เวกเตอร์ผลลัพธ์ (\mathbf{o}_k) ที่ได้จากขั้นตอนที่ 2 นำไปหาตำแหน่งที่ค่าในเวกเตอร์ผลลัพธ์มีค่ามากที่สุด เพื่อหาว่าเวกเตอร์ข้อมูลควรจะเป็นสมาชิกในคลาสใด เช่น สมมุติว่าตำแหน่งที่ k^{th} ของเวกเตอร์ผลลัพธ์มีค่ามากที่สุด หมายความว่าเวกเตอร์ข้อมูลควรจะเป็นสมาชิกในคลาส k^{th} ดังสมการที่ 3.36

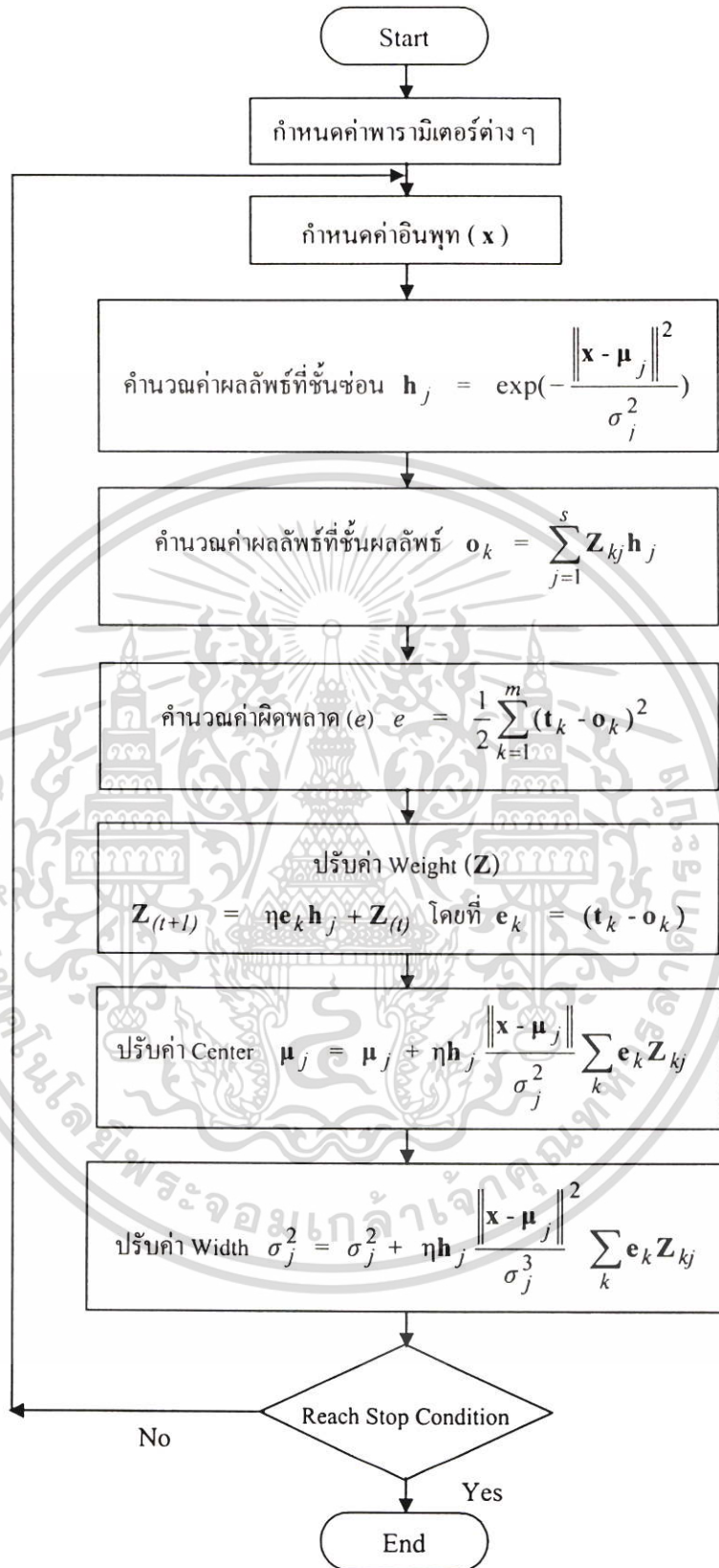
$$\mathbf{x} \in \omega_k \quad \text{if } k^{\text{th}} = \max(\mathbf{o}_k) \quad \text{for } k = 1, 2, \dots, m \quad (3.36)$$

โดยที่

- m คือ มิติของเวกเตอร์ผลลัพธ์ (จำนวน โหนดในชั้นผลลัพธ์)
- k^{th} คือ ตำแหน่งที่มีค่ามากที่สุด ในเวกเตอร์ผลลัพธ์
- ω_k คือ คลาสที่เวกเตอร์ข้อมูลเป็นสมาชิก (Winner Class)

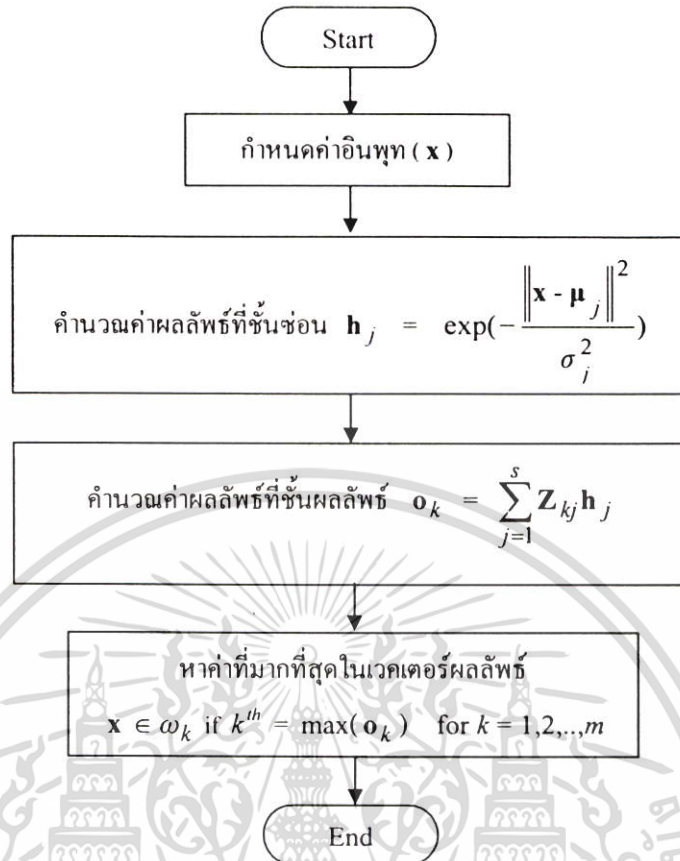
ซึ่งสามารถสรุปขั้นตอนการรู้จำของโครงข่ายได้ ดังรูปที่ 3.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 ไดอะแกรมแสดงขั้นตอนการเรียนรู้ของโครงข่ายประสาทเทียมแบบ RBF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 ไดอะแกรมแสดงขั้นตอนการรู้จำของโครงข่ายประสาทเทียมแบบ RBF

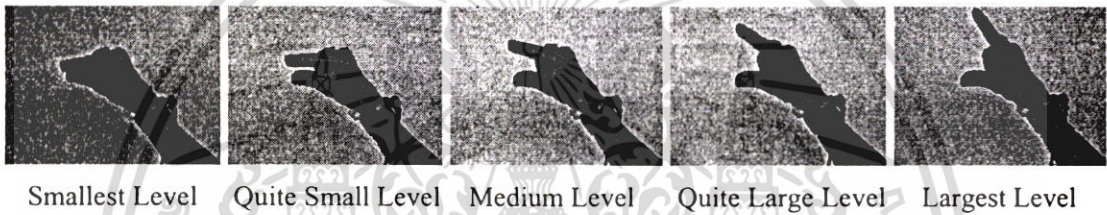
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การพัฒนาโครงข่ายประสาทเทียม สำหรับการรู้จำท่ามือบอกขนาดหรือปริมาณ

4.1 แนวคิดเบื้องต้น

จากการศึกษาท่ามือภาษาไทย (Thai Sign Language) [12] พบว่ามีท่ามือพื้นฐานบางท่าเป็นท่าบอกขนาดหรือปริมาณ (Quantitative Posture) กล่าวคือ ท่ามือเหล่านี้จะมีความหมายในแง่ของการบอกขนาดหรือบอกปริมาณที่เฉพาะเจาะจงว่ามีขนาดหรือปริมาณเท่าใด ยกตัวอย่างเช่น ท่ามือที่มีความหมายว่า “ความหนาของวัตถุ” ดังแสดงในรูปที่ 4.1



รูปที่ 4.1 ท่ามือบอกขนาดหรือปริมาณที่มีความหมายว่า “ความหนาของวัตถุ”

การใช้โครงข่ายประสาทเทียมแบบ Standard Multi-Layer Perceptron (SMLP) [8-10] ในการรู้จำท่ามือบอกขนาดหรือปริมาณเหล่านี้ทำให้เกิดปัญหา ซึ่งสรุปได้ดังนี้

- ต้องใช้จำนวนข้อมูลในการเรียนรู้ของโครงข่ายจำนวนมาก กล่าวคือ โครงข่ายประสาทเทียมแบบ SMLP จะต้องใช้จำนวนข้อมูลของท่ามือบอกขนาดหรือปริมาณทุกระดับที่ต้องการให้โครงข่ายสามารถจำแนกได้ ยกตัวอย่างเช่น จากรูปที่ 4.1 มีจำนวนระดับขนาดหรือปริมาณ (Quantitative Level) ทั้งหมด 5 ระดับ ดังนั้น จะต้องใช้ข้อมูลทั้ง 5 ระดับในการเรียนรู้ของโครงข่าย
- ความซับซ้อนในการคำนวณของโครงข่ายประสาทเทียมแบบ Standard Multi-Layer Perceptron ในขั้นตอนของการรู้จำจะเพิ่มขึ้นอย่างมาก ถ้าจำนวนท่ามือบอกขนาดหรือปริมาณ และจำนวนระดับขนาดหรือปริมาณ (Quantitative Level) ที่ต้องการรู้จำเพิ่มมากขึ้น

ดังนั้น ในงานวิจัยฉบับนี้ จึงได้นำเสนอวิธีการพัฒนาโครงข่ายประสาทเทียมสำหรับการรู้จำท่ามือบอกขนาดหรือปริมาณเพื่อแก้ปัญหาที่กล่าวมาข้างต้น

4.2 การพัฒนาโครงข่ายประสาทเทียมสำหรับรู้จำท่ามือบอกขนาดหรือปริมาณ

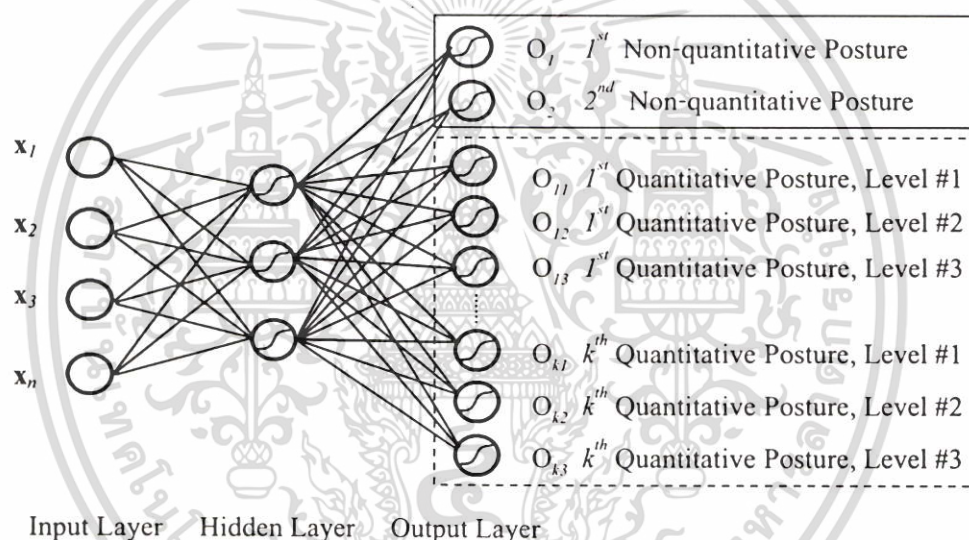
ในส่วนนี้ จะกล่าวถึงการพัฒนาโครงข่ายประสาทเทียม เพื่อการรู้จำท่ามือบอกขนาดหรือปริมาณ ในงานวิจัยฉบับนี้จะนำเสนอวิธีการในการพัฒนาทั้งหมด 2 แบบ โดยให้ชื่อโครงข่ายที่พัฒนานี้ว่า Quantitative-classification Multi-Layer Perceptron (QMLP)

4.2.1 โครงสร้างของโครงข่าย

ในส่วนนี้จะกล่าวถึงโครงสร้างของโครงข่ายประสาทเทียมแบบ SMLP เปรียบเทียบกับโครงสร้างของโครงข่ายแบบ QMLP ในการรู้จำท่ามือบอกขนาดหรือปริมาณ

4.2.1.1 Standard Multi-Layer Perceptron (SMLP)

โครงสร้างของโครงข่ายประสาทเทียมแบบ Standard Multi-Layer Perceptron (SMLP) [8-10] สำหรับการรู้จำท่ามือบอกขนาดหรือปริมาณ แสดงได้ดังรูปที่ 4.2

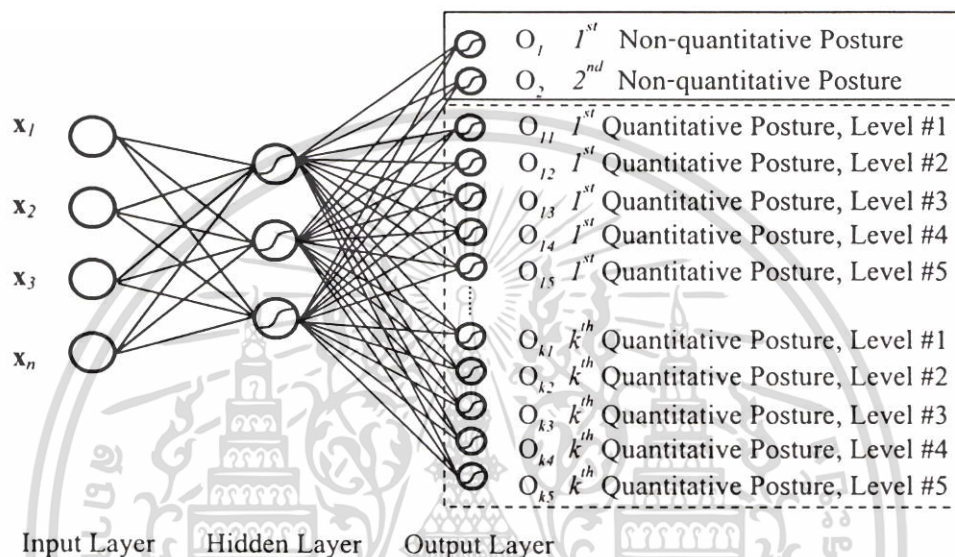


รูปที่ 4.2 โครงข่ายประสาทเทียมแบบ SMLP สำหรับรู้จำท่ามือบอกขนาดหรือปริมาณ กรณีที่ท่ามือบอกขนาดหรือปริมาณต้องการรู้จำขนาดหรือปริมาณ 3 ระดับ

จากรูปที่ 4.2 จะพบว่า โครงสร้างของโครงข่ายประสาทเทียมแบบ SMLP สำหรับรู้จำท่ามือบอกขนาดหรือปริมาณ จะมีลักษณะ คือ ที่ชั้นผลลัพธ์ (Output Layer) จะประกอบไปด้วยกลุ่มโหนด (Node) ทั้งหมด 2 กลุ่ม คือ กลุ่มแรก เป็นกลุ่มโหนดของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (ภายในสี่เหลี่ยมเส้นทึบ) จากรูปคือ O_1 และ O_2 ซึ่ง O_1 คือ โหนดของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณท่าที่ 1st ส่วน O_2 คือ โหนดของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณท่าที่ 2nd ส่วนกลุ่มโหนดที่สอง คือ กลุ่มโหนดของท่ามือบอกขนาดหรือปริมาณ (ภายในสี่เหลี่ยมเส้นประ) จากรูป ท่ามือบอกขนาดหรือปริมาณแต่ละท่ามีจำนวนระดับขนาด (Quantitative Level)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ต้องการรู้จำท่ามือละ 3 ระดับ ดังนั้น ในท่ามือบอกขนาดหรือปริมาณในแต่ละท่าจะต้องมีโหนดทั้งหมด 3 โหนด แต่ละโหนดก็จะแทนระดับขนาดแต่ละระดับ อาจจะกล่าวได้ว่าแต่ละระดับขนาดจะถูกกำหนดให้เป็น 1 ท่ามือ (1 คลาส) ยกตัวอย่างเช่น โหนด O_{11} , O_{12} และ O_{13} คือ โหนดที่แทนท่ามือบอกขนาดหรือปริมาณท่าที่ 1^{st} ระดับขนาดที่ 1, 2 และ 3 ตามลำดับ ส่วนโหนด O_{k1} , O_{k2} และ O_{k3} คือ โหนดที่แทนท่ามือบอกขนาดหรือปริมาณท่าที่ k^{th} ระดับขนาดที่ 1, 2 และ 3 ตามลำดับ ซึ่งที่ชั้นซ่อนและชั้นผลลัพธ์ของโครงข่ายจะใช้ Activate Function คือ Sigmoid Function



รูปที่ 4.3 โครงข่ายประสาทเทียมแบบ SMLP สำหรับรู้จำท่ามือบอกขนาดหรือปริมาณ กรณีที่ท่ามือบอกขนาดหรือปริมาณต้องการรู้จำขนาดหรือปริมาณ 5 ระดับ

จากรูปที่ 4.3 เป็นรูปที่แสดงโครงสร้างของโครงข่ายประสาทเทียมแบบ SMLP กรณีที่ท่ามือบอกขนาดหรือปริมาณแต่ละท่าต้องการรู้จำระดับขนาดหรือปริมาณทั้งหมด 5 ระดับ ดังนั้นท่ามือบอกขนาดหรือปริมาณแต่ละท่าจะมีโหนดที่เกี่ยวข้องของท่ามือละ 5 โหนด ยกตัวอย่างเช่น โหนด O_{11} , O_{12} , O_{13} , O_{14} และ O_{15} คือ โหนดที่แทนท่ามือบอกขนาดหรือปริมาณท่าที่ 1^{st} ระดับขนาดที่ 1, 2, 3, 4 และ 5 ตามลำดับ ส่วนโหนด O_{k1} , O_{k2} , O_{k3} , O_{k4} และ O_{k5} คือ โหนดที่แทนท่ามือบอกขนาดหรือปริมาณท่าที่ k^{th} ระดับขนาดที่ 1, 2, 3, 4 และ 5 ตามลำดับ ซึ่งจะพบว่าโครงสร้างของโครงข่ายประสาทเทียมแบบ SMLP จะเปลี่ยนไปเมื่อจำนวนระดับขนาดหรือปริมาณที่ต้องการรู้จำเปลี่ยนไป

จำนวนโหนดทั้งหมดในชั้นผลลัพธ์สามารถคำนวณได้ดังสมการที่ 4.1

$$\text{Output Node} = \text{NonQP} + (\text{QP} \times \text{Level}) \quad (4.1)$$

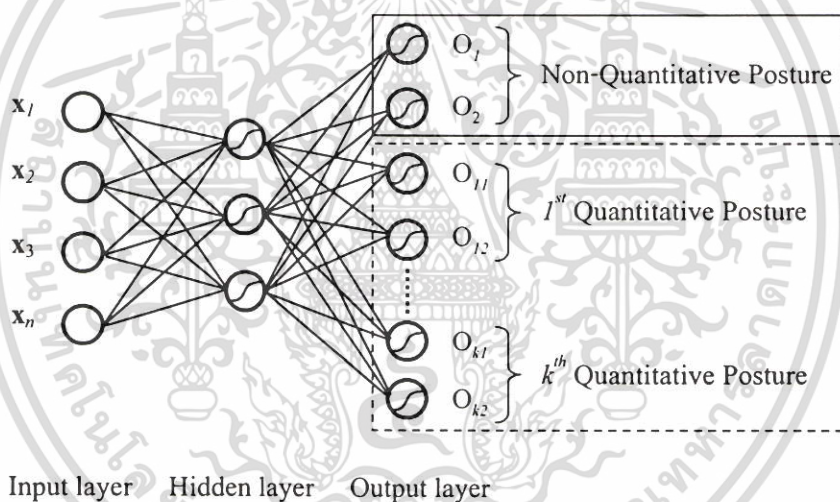
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่

Output Node	คือ จำนวนโหนดที่ชั้นผลลัพธ์ หรือขนาดของเวกเตอร์ผลลัพธ์
NonQP	คือ จำนวนของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (Non-Quantitative Posture) ทั้งหมด
QP	คือ จำนวนของท่ามือบอกขนาดหรือปริมาณทั้งหมด
Level	คือ จำนวนระดับขนาด (Quantitative Level) ของท่ามือบอกขนาดหรือปริมาณ

4.2.1.2 Quantitative-classification Multi-Layer Perceptron – Type I (QMLP-I)

โครงสร้างของ โครงข่ายประสาทเทียมแบบ Quantitative-classification Multi-Layer Perceptron - Type I (QMLP-I) แสดงได้ดังรูปที่ 4.4



รูปที่ 4.4 โครงข่ายประสาทเทียมแบบ QMLP-I สำหรับรู้จำท่ามือบอกขนาดหรือปริมาณ

จากรูปที่ 4.4 พบว่าที่ชั้นผลลัพธ์ (Output Layer) จะประกอบไปด้วยกลุ่มโหนด (Node) ทั้งหมด 2 กลุ่ม คือ กลุ่มแรก เป็นกลุ่มโหนดของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (ภายในสี่เหลี่ยมเส้นทึบ) จากรูปคือ O_1 และ O_2 ซึ่งคือ โหนดของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ ท่าที่ 1st และ 2nd ตามลำดับ กลุ่มโหนดที่สอง คือ กลุ่มโหนดของท่ามือบอกขนาดหรือปริมาณ (ภายในสี่เหลี่ยมเส้นประ) ซึ่งในส่วนนี้จะมีความแตกต่างจาก โครงข่ายแบบ SMLP คือ ท่ามือบอกขนาดหรือปริมาณแต่ละท่าจะมีโหนดเพียง 2 โหนดเท่านั้น ไม่ว่าจำนวนระดับขนาด (Quantitative Level) ที่ต้องการรู้จำจะมีกี่ระดับก็ตาม ยกตัวอย่างเช่น O_{11} และ O_{12} เป็นโหนดที่ 1 และ 2 ของท่ามือบอกขนาดหรือปริมาณท่าที่ 1st ส่วน O_{k1} และ O_{k2} เป็นโหนดที่ 1 และ 2 ของท่ามือบอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขนาดท่าที่ k^{th} ซึ่งทั้ง 2 โหนดของท่ามือบอกขนาดแต่ละท่า จะนำไปใช้ในการระบุความหมายและระบุขนาดของท่ามือบอกขนาดหรือปริมาณ ซึ่งจะได้กล่าวอย่างละเอียดต่อไป ซึ่งที่ชั้นซ่อนและชั้นผลลัพธ์ของโครงข่ายจะใช้ Activate Function คือ Sigmoid Function จำนวนโหนดทั้งหมดในชั้นผลลัพธ์สามารถคำนวณได้ดังสมการที่ 4.2

$$\text{Output Node} = \text{NonQP} + (2 \cdot \text{QP}) \quad (4.2)$$

โดยที่

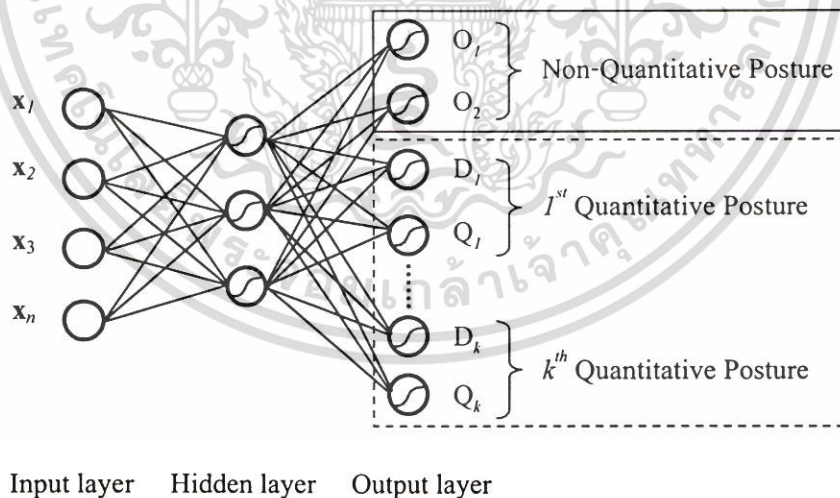
Output Node คือ จำนวนโหนดที่ชั้นผลลัพธ์ หรือขนาดของเวกเตอร์ผลลัพธ์

NonQP คือ จำนวนของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (Non-Quantitative Posture) ทั้งหมด

QP คือ จำนวนของท่ามือบอกขนาดหรือปริมาณทั้งหมด

4.2.1.3 Quantitative-classification Multi-Layer Perceptron – Type II (QMLP-II)

โครงสร้างของโครงข่ายประสาทเทียมแบบ Quantitative-classification Multi-Layer Perceptron - Type II (QMLP-II) แสดงได้ดังรูปที่ 4.5



รูปที่ 4.5 โครงข่ายประสาทเทียมแบบ QMLP-II สำหรับรู้จำท่ามือบอกขนาดหรือปริมาณ

จากรูปที่ 4.5 พบว่าโครงสร้างของโครงข่ายประสาทเทียมแบบ QMLP-II มีลักษณะที่คล้ายคลึงกับโครงข่ายประสาทเทียมแบบ QMLP-I กล่าวคือ ที่ชั้นผลลัพธ์จะประกอบไปด้วยกลุ่มโหนดทั้งหมด 2 กลุ่ม คือ กลุ่มแรก เป็นกลุ่มโหนดของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (ภายในสี่เหลี่ยมเส้นทึบ) จากรูป O_1, O_2 คือ โหนดของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณท่าที่ 1^{st} และ 2^{nd} ตามลำดับ ส่วนกลุ่มโหนดที่สอง คือ กลุ่มโหนดของท่ามือบอกขนาด หรือปริมาณ (ภายในสี่เหลี่ยมเส้นประ) ท่ามือบอกขนาดหรือปริมาณแต่ละท่า ก็จะมีโหนดที่เกี่ยวข้องทั้งหมด 2 โหนดเหมือนกับโครงข่ายแบบ QMLP-I แต่จะมีความแตกต่างกัน คือ ผลลัพธ์ที่ได้ในแต่ละโหนดของท่ามือบอกขนาดหรือปริมาณ จะมีวัตถุประสงค์ในการนำไปใช้ที่แตกต่างกัน กล่าวคือ โหนดที่ 1 (D_1) ของท่ามือบอกขนาดหรือปริมาณท่าที่ 1^{st} ผลลัพธ์ของโหนดนี้จะนำไปเปรียบเทียบกับค่าผลลัพธ์ของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (จากรูปคือ O_1 และ O_2) เพื่อหาความหมายของเวกเตอร์ข้อมูลที่ต้องการรู้จำ สำหรับโหนดที่ 2 (Q_1) ของท่ามือบอกขนาดหรือปริมาณท่าที่ 1^{st} ซึ่งผลลัพธ์ของโหนดนี้จะนำไปใช้ในการระบุขนาดของเวกเตอร์ข้อมูล ในขณะที่เวกเตอร์ข้อมูลเป็นท่ามือบอกขนาดหรือปริมาณ

4.3.2 การกำหนดค่าผลลัพธ์ที่ต้องการ (Desired Output) ให้กับข้อมูล

การกำหนดค่าผลลัพธ์ที่ต้องการ (Desired Output) ให้กับข้อมูลที่จะนำมาใช้ในการเรียนรู้ (Learning) ของโครงข่ายประสาทเทียมแบบ SMLP กับโครงข่ายประสาทเทียมแบบ QMLP จะเป็นดังนี้

4.3.2.1 Standard Multi-Layer Perceptron (SMLP)

เนื่องจากว่าระดับขนาดแต่ละระดับของท่ามือบอกขนาดหรือปริมาณ จะถูกกำหนดให้เป็น 1 ท่ามือ (1 คลาส) ซึ่งจะมีโหนดที่ชั้นผลลัพธ์เป็นของตัวเอง ดังนั้น การกำหนดค่าผลลัพธ์ที่ต้องการให้กับโครงข่ายประสาทเทียมแบบ SMLP จะมีวิธีการดังนี้

กำหนดให้

x_i คือ เวกเตอร์ข้อมูลที่เป็นสมาชิกของคลาส i^{th}

y คือ เวกเตอร์ผลลัพธ์ของ x_i

ดังนั้น ค่าผลลัพธ์ที่ต้องการของเวกเตอร์ข้อมูล x_i กำหนดได้ดังนี้

$$y_{k^{th}} = 0.9; \text{ if } k = i \text{ และ } y_{k^{th}} = 0.1; \text{ if } k \neq i; \text{ for } k = 1 \dots m \quad (4.3)$$

โดยที่

m คือ จำนวนโหนดทั้งหมดที่ชั้นผลลัพธ์

จากสมการที่ 4.3 สามารถอธิบายได้ว่า สำหรับข้อมูลของท่ามือ ท่าที่ i^{th} ค่าผลลัพธ์ที่ต้องการ ณ ตำแหน่ง i^{th} ในที่นี้กำหนดให้ค่าผลลัพธ์เท่ากับ 0.9 ส่วนค่าผลลัพธ์ ณ ตำแหน่งอื่น ๆ กำหนดให้มีค่าเท่ากับ 0.1

4.3.2.2 Quantitative-classification Multi-Layer Perceptron – Type I (QMLP-I)

ในการกำหนดค่าผลลัพธ์ที่ต้องการให้กับท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณกับท่ามือที่เป็นท่าบอกขนาดหรือปริมาณของโครงข่ายประสาทเทียมแบบ QMLP-I จะมีวิธีการกำหนดที่แตกต่างกัน ดังนี้คือ

- ท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (Non-Quantitative Posture)

กำหนดให้

x_i คือ เวกเตอร์ข้อมูลที่เป็นท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ ท่าที่ i^{th}

y คือ เวกเตอร์ผลลัพธ์ของ x_i

ดังนั้น ค่าผลลัพธ์ที่ต้องการของเวกเตอร์ข้อมูล x_i กำหนดได้ดังนี้

$$y_{k^{th}} = 0.9; \text{ if } k = i \text{ และ } y_{k^{th}} = 0.1; \text{ if } k \neq i; \text{ for } k = 1 \dots m \quad (4.4)$$

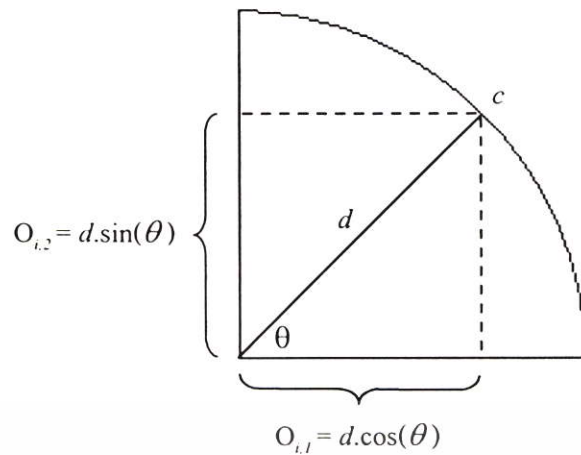
โดยที่

m คือ จำนวนโหนดทั้งหมดที่ชั้นผลลัพธ์

จากสมการที่ 4.4 สามารถอธิบายได้ว่า สำหรับข้อมูลของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ ท่าที่ i^{th} ค่าผลลัพธ์ที่ต้องการ ณ ตำแหน่ง i^{th} ในที่นี้กำหนดให้ค่าผลลัพธ์เท่ากับ 0.9 ส่วนค่าผลลัพธ์ ณ ตำแหน่งอื่น ๆ กำหนดให้มีค่าเท่ากับ 0.1

- ท่ามือที่เป็นท่าบอกขนาดหรือปริมาณ (Quantitative Posture)

เนื่องจากว่าท่ามือบอกขนาดหรือปริมาณแต่ละท่า จะมีโหนดที่ชั้นผลลัพธ์ท่ามือละ 2 โหนด ซึ่งการกำหนดค่าผลลัพธ์ที่ต้องการให้กับท่ามือเหล่านี้ จะอาศัยรูปที่ 4.6 ช่วยในการกำหนดค่าผลลัพธ์ที่ต้องการให้กับท่ามือบอกขนาดหรือปริมาณ



รูปที่ 4.6 การกำหนดค่าผลลัพธ์ที่ต้องการให้กับข้อมูลของท่ามือบอกขนาดหรือปริมาณของวิธีการแบบ QMLP-I

รูปที่ 4.6 มีตัวแปรที่น่าสนใจ ดังนี้คือ

$O_{1,1}$ คือ โหนดที่ 1 ของท่ามือบอกขนาดหรือปริมาณที่ i^{th}

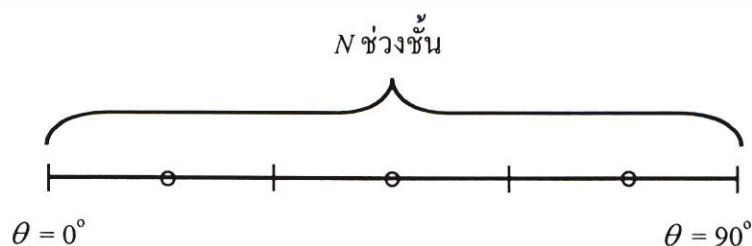
$O_{1,2}$ คือ โหนดที่ 2 ของท่ามือบอกขนาดหรือปริมาณที่ i^{th}

c คือ จุดที่ค่าผลลัพธ์ทั้ง 2 โหนดมีความสัมพันธ์กัน

d คือ ค่าระยะทางจากจุด $(0,0)$ ไปยังจุด c ใช้ในการระบุความหมายของข้อมูล ในที่นี้กำหนดให้มีค่าเท่ากับ 0.9

θ คือ มุมที่จุด c ทำกับแกน x ใช้ในการระบุขนาดหรือปริมาณของท่ามือบอกขนาดหรือปริมาณ มีค่าระหว่าง $0 - 90$ องศา

ในการกำหนดค่าผลลัพธ์ที่ต้องการให้กับข้อมูลของท่ามือบอกขนาดหรือปริมาณ ขั้นตอนแรกคือ การแบ่งค่า θ (0 ถึง 90 องศา) ออกเป็นช่วงชั้นที่เท่า ๆ กัน ทั้งหมด N ช่วงชั้น ซึ่งค่า N จะเป็นตัวที่ควบคุมจำนวนระดับขนาดหรือปริมาณ ดังรูปที่ 4.7



รูปที่ 4.7 การกำหนดระดับขนาดให้กับท่ามือบอกขนาดหรือปริมาณของโครงข่ายแบบ QMLP-I

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.7 จะมีการแบ่งค่า θ เป็น 3 ช่วงชั้น ($N = 3$) เนื่องจาก ท่ามือบอกขนาดหรือปริมาณแต่ละท่า จะใช้จำนวนระดับขนาดหรือปริมาณในขั้นตอนของการเรียนรู้เพียง 3 ระดับเท่านั้น ซึ่งแต่ละช่วงชั้นก็จะแทนขนาดหรือปริมาณแต่ละระดับที่นำมาใช้ในขั้นตอนของการเรียนรู้ของโครงข่าย การกำหนดค่าผลลัพธ์ที่ต้องการให้กับทั้ง 2 โหนด ($O_{1,1}$ และ $O_{1,2}$ จากรูปที่ 4.6) ของท่ามือบอกขนาดหรือปริมาณแต่ละท่า เมื่อแปลงค่าผลลัพธ์ทั้ง 2 โหนดแล้ว สามารถนำไปใช้รู้จำความหมายและรู้จำขนาดหรือปริมาณของเวกเตอร์ข้อมูล กำหนดได้โดยการนำค่าองศาที่เป็นค่ากึ่งกลางของช่วงชั้นที่สอดคล้องกับข้อมูลที่ต้องการกำหนดค่า ไปแทนค่าในสมการที่ 4.5 และ 4.6

$$O_{1,1} = d \cdot \cos(\theta) \quad (4.5)$$

$$O_{1,2} = d \cdot \sin(\theta) \quad (4.6)$$

4.3.2.3 Quantitative-classification Multi-Layer Perceptron – Type II (QMLP-II)

ในการกำหนดค่าผลลัพธ์ที่ต้องการให้กับโครงข่ายประสาทเทียมแบบ QMLP-II จะมีวิธีการ ดังนี้คือ

- ท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (Non-Quantitative Posture)

กำหนดให้

x_i คือ เวกเตอร์ข้อมูลที่เป็นท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ ท่าที่ i^{th}

y คือ เวกเตอร์ผลลัพธ์ของ x_i

ดังนั้น ค่าผลลัพธ์ที่ต้องการของเวกเตอร์ข้อมูล x_i กำหนดได้ดังนี้

$$y_{k^{th}} = 0.9; \text{ if } k = i \text{ และ } y_{k^{th}} = 0.1; \text{ if } k \neq i; \text{ for } k = 1 \dots m \quad (4.7)$$

โดยที่

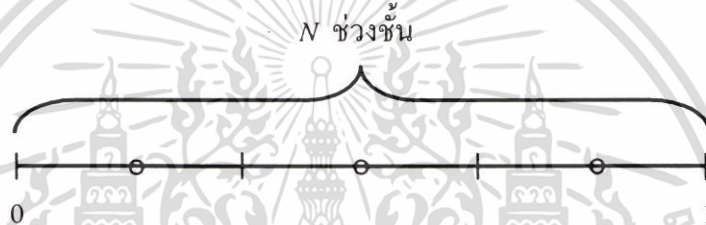
m คือ จำนวน โหนดทั้งหมดที่ชั้นผลลัพธ์

จากสมการที่ 4.7 สามารถอธิบายได้ว่า สำหรับข้อมูลของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ ท่าที่ i^{th} ค่าผลลัพธ์ที่ต้องการ ณ ตำแหน่ง i^{th} ในที่นี้กำหนดให้ค่าผลลัพธ์เท่ากับ 0.9 ส่วนค่าผลลัพธ์ ณ ตำแหน่งอื่น ๆ กำหนดให้มีค่าเท่ากับ 0.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ท่ามือที่เป็นท่าบอกขนาดหรือปริมาณ (Quantitative Posture)

เนื่องจากว่าท่ามือบอกขนาดหรือปริมาณแต่ละท่า จะมีโหนดที่ชั้นผลลัพธ์ท่ามือละ 2 โหนด ซึ่งการกำหนดค่าผลลัพธ์ที่ต้องการให้กับท่ามือบอกขนาดหรือปริมาณ มีวิธีการคือ สำหรับข้อมูลของท่ามือบอกขนาดหรือปริมาณ ท่าที่ i^{th} ค่าผลลัพธ์ที่ต้องการของโหนดที่ 1 (D_i) ที่ใช้ในการระบุความหมายของเวกเตอร์ข้อมูล ในที่นี้กำหนดให้ค่าผลลัพธ์เท่ากับ 0.9 ส่วนค่าผลลัพธ์ตำแหน่งอื่น ๆ กำหนดให้มีค่าเท่ากับ 0.1 ส่วนโหนดที่สองของท่ามือบอกขนาดหรือปริมาณ (Q_i) ซึ่งเป็นโหนดที่ใช้ในการระบุขนาดของท่ามือ สามารถกำหนดค่าได้โดยการแบ่งค่าผลลัพธ์ของ Activate Function ที่ใช้ในชั้นผลลัพธ์ ในที่นี้ใช้ Sigmoid Function ซึ่งมีค่าระหว่าง 0 – 1 ออกเป็นช่วงชั้นที่เท่า ๆ กัน ทั้งหมด N ช่วงชั้น ซึ่งค่า N จะเป็นตัวที่ควบคุมจำนวนระดับขนาด หรือ ปริมาณ ดังรูปที่ 4.8



รูปที่ 4.8 การกำหนดระดับขนาดให้กับท่ามือบอกขนาดหรือปริมาณของโครงข่ายแบบ QMLP-II

จากรูปที่ 4.8 จะมีการแบ่งค่าผลลัพธ์ของ Sigmoid Function เป็น 3 ช่วงชั้น ($N = 3$) เนื่องจาก ท่ามือบอกขนาดหรือปริมาณแต่ละท่า จะใช้จำนวนระดับขนาดหรือปริมาณในขั้นตอนของการเรียนรู้เพียง 3 ระดับ เท่านั้น ซึ่งแต่ละช่วงชั้นก็จะแทนขนาดหรือปริมาณแต่ละระดับที่นำมาใช้ในขั้นตอนของการเรียนรู้ของโครงข่าย การกำหนดค่าผลลัพธ์ที่ต้องการให้กับโหนดที่ 2 (Q_i) กำหนดได้โดยการนำค่าที่เป็นค่ากึ่งกลางของช่วงชั้นที่สอดคล้องกับข้อมูล ไปกำหนดเป็นค่าผลลัพธ์ที่ต้องการให้กับโหนดที่ 2

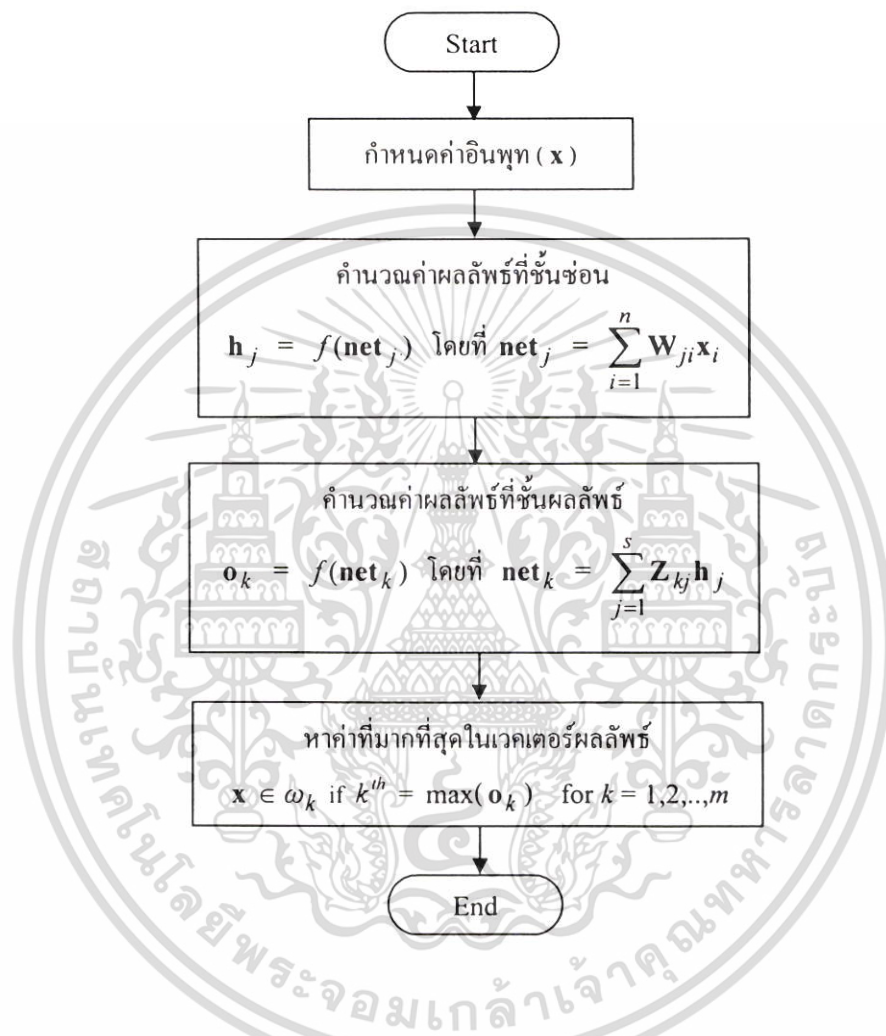
4.3.3 การเรียนรู้ของโครงข่าย

หลังจากที่ได้กำหนดค่าผลลัพธ์ที่ต้องการให้กับข้อมูลแล้ว ก็จะนำข้อมูลเหล่านั้นมาใช้ในการเรียนรู้โครงข่าย ซึ่งการเรียนรู้ของโครงข่ายประสาทเทียมแบบ QMLP ทั้ง 2 แบบ จะใช้ back-propagation algorithm ในการเรียนรู้เหมือนกับการเรียนของโครงข่ายประสาทเทียมแบบ SMLP

4.3.4 การรู้จำของโครงข่าย

4.3.4.1 Standard Multi-Layer Perceptron (SMLP)

การรู้จำของโครงข่ายประสาทเทียมแบบ SMLP มีขั้นตอนของการรู้จำซึ่งสามารถสรุปได้ดังรูปที่ 4.9



รูปที่ 4.9 ไคอะแกรมแสดงขั้นตอนการรู้จำทำมือบอกขนาดหรือปริมาณของโครงข่ายแบบ SMLP

4.3.4.2 Quantitative-classification Multi-Layer Perceptron – Type I (QMLP-I)

การรู้จำทำมือของโครงข่ายแบบ QMLP-I แบ่งเป็น 2 ขั้นตอน คือ

1. การรู้จำความหมาย
2. การรู้จำขนาดหรือปริมาณ ในกรณีที่ข้อมูลที่ต้องการรู้จำเป็นทำมือบอกขนาดหรือปริมาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การรู้จำความหมาย

เนื่องจากว่าท่ามือบอกขนาดหรือปริมาณแต่ละท่า จะมีโหนดที่ชั้นผลลัพธ์ทั้งหมด 2 โหนด ดังนั้นจะต้องมีการแปลงค่าทั้ง 2 โหนดนี้ ให้เหลือเพียงค่าเดียว (\hat{d}_i) ซึ่งสามารถคำนวณได้ ดังสมการที่ 4.8

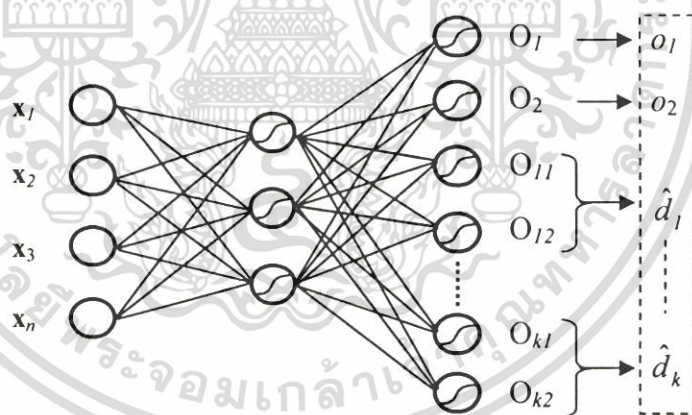
$$\hat{d}_i = \sqrt{(o_{i,1})^2 + (o_{i,2})^2} \quad (4.8)$$

โดยที่

$o_{i,1}$ คือ ค่าผลลัพธ์ของโหนดที่ 1 ของท่ามือบอกขนาดหรือปริมาณท่าที่ i^{th}

$o_{i,2}$ คือ ค่าผลลัพธ์ของโหนดที่ 2 ของท่ามือบอกขนาดหรือปริมาณท่าที่ i^{th}

ซึ่งค่า \hat{d}_i ที่ได้จากสมการที่ 4.8 จะเป็นค่าที่นำไปเปรียบเทียบกับค่าผลลัพธ์ของท่ามือที่ไม่ใช่ท่ามือบอกขนาดหรือปริมาณ เพื่อหาดำแหน่งที่มีค่ามากที่สุด ซึ่งเรียกว่า ผู้ชนะ (Winner Class) กล่าวคือ ถ้าหากตำแหน่งใดมีค่ามากที่สุด ก็แสดงว่าข้อมูลที่ต้องการรู้จำเป็นสมาชิกของท่ามือนั้น ซึ่งแสดงได้ดังรูปที่ 4.10



รูปที่ 4.10 แสดงการรู้จำความหมายของโครงข่ายประสาทเทียมแบบ QMLP-I

2. การรู้จำขนาดหรือปริมาณ

ในกรณีที่ท่ามือบอกขนาดหรือปริมาณ i^{th} เป็นท่าที่มีค่าผลลัพธ์มากที่สุด (Winner Class) จะต้องมีการคำนวณหาขนาดหรือปริมาณของท่ามือนั้นด้วย โดยใช้สมการที่ 4.9

$$\hat{\theta}_i = \tan^{-1}(o_{i,2}/o_{i,1}) \quad (4.9)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

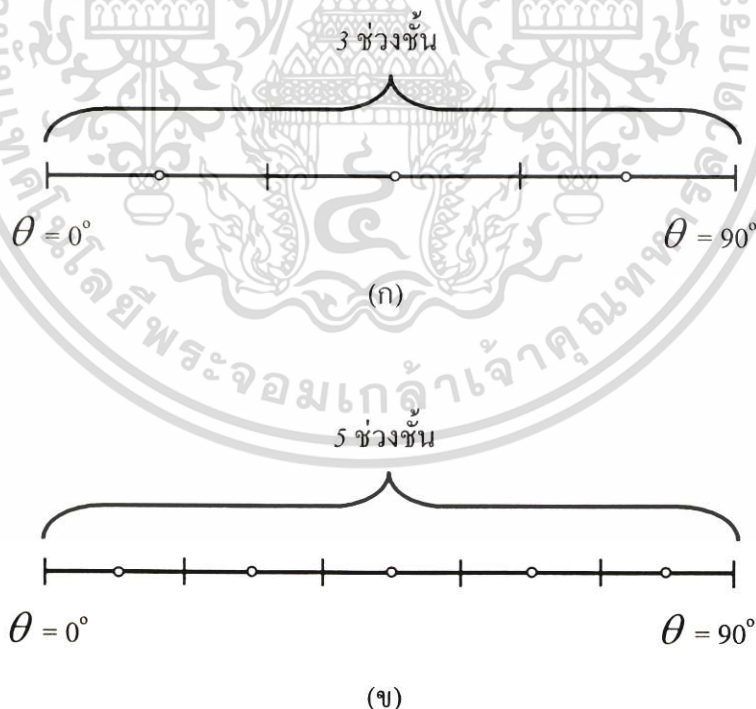
โดยที่

$\hat{\theta}_i$ คือ ค่าที่ใช้ในการระบุขนาดของท่ามือบอกขนาดหรือปริมาณท่าที่ i^{th}

$o_{i,1}$ คือ ค่าผลลัพธ์ของโหนดที่ 1 ($O_{i,1}$) ของท่ามือบอกขนาดหรือปริมาณท่าที่ i^{th} (Winner Class)

$o_{i,2}$ คือ ค่าผลลัพธ์ของโหนดที่ 2 ($O_{i,2}$) ของท่ามือบอกขนาดหรือปริมาณท่าที่ i^{th} (Winner Class)

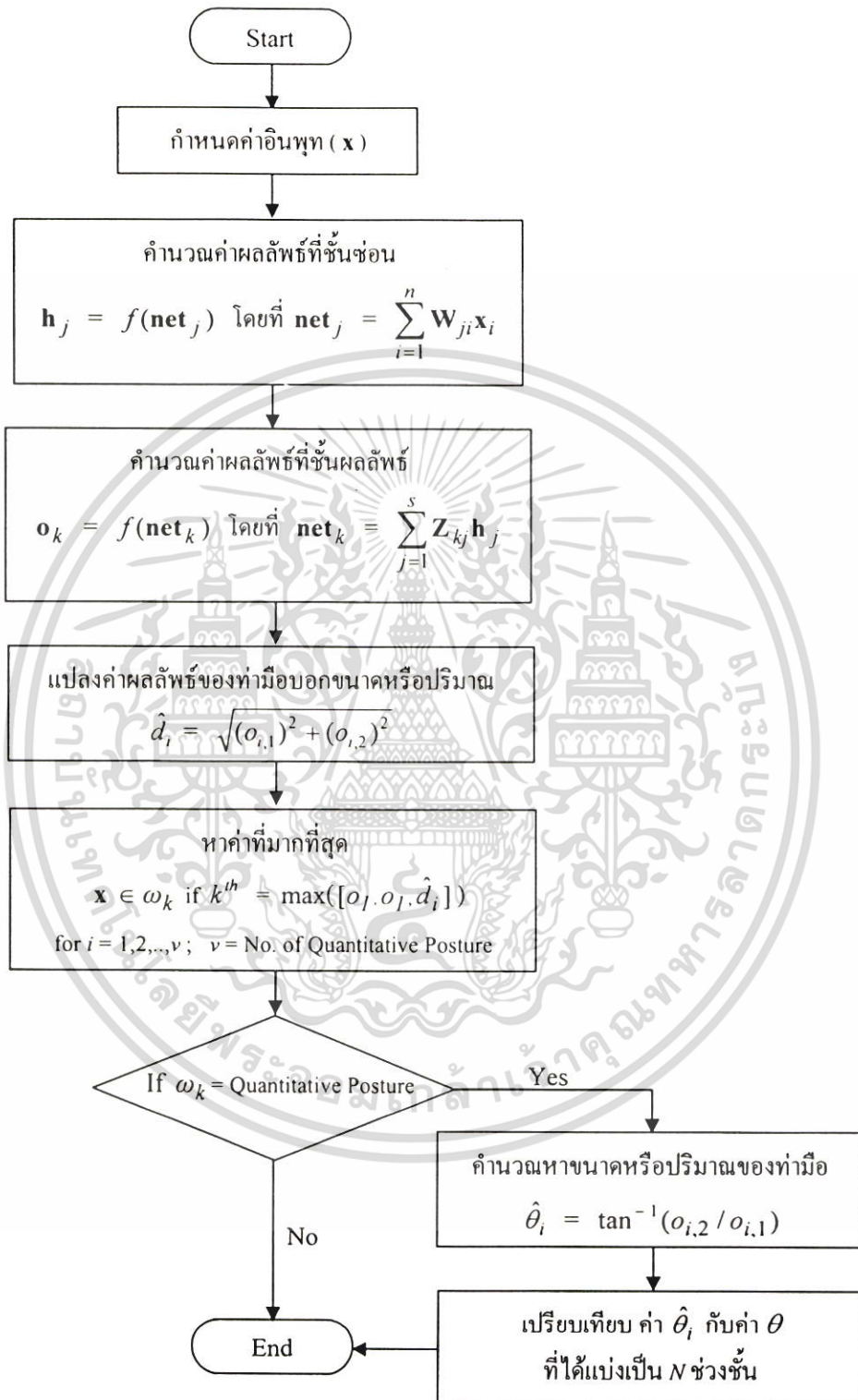
ในการระบุขนาดของท่ามือทำได้โดยการแบ่งค่าองศาออกเป็น N ช่วงชั้น ขึ้นอยู่กับระดับขนาดหรือปริมาณที่ต้องการรู้จำ ยกตัวอย่างเช่น ถ้าหากว่าท่ามือบอกขนาดหรือปริมาณแต่ละท่าต้องการรู้จำขนาดหรือปริมาณทั้งหมด 3 ระดับ ก็ต้องแบ่งองศาออกเป็น 3 ช่วงชั้น หลังจากนั้นนำค่าผลลัพธ์ ($\hat{\theta}_i$) ที่ได้จากสมการที่ 4.9 ไปเปรียบเทียบกับค่าผลลัพธ์ ($\hat{\theta}_i$) อยู่ในช่วงชั้นใด แสดงว่าเวกเตอร์ข้อมูลมีขนาดหรือปริมาณระดับนั้น รูปที่ 4.11 (ก) ถ้าหากว่าท่ามือบอกขนาดหรือปริมาณแต่ละท่าต้องการรู้จำขนาดหรือปริมาณทั้งหมด 5 ระดับ ก็ต้องแบ่งองศาออกเป็น 5 ช่วงชั้น แล้วก็นำค่าผลลัพธ์ ($\hat{\theta}_i$) ที่ได้จากสมการที่ 4.9 ไปเปรียบเทียบกับค่าผลลัพธ์ ($\hat{\theta}_i$) อยู่ในช่วงชั้นใด แสดงว่าเวกเตอร์ข้อมูลมีขนาดหรือปริมาณระดับนั้น รูปที่ 4.11 (ข)



รูปที่ 4.11 แสดงรู้จำขนาดหรือปริมาณของโครงข่ายประสาทเทียมแบบ QMLP-I

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนการรู้จำของโครงข่ายประสาทเทียมแบบ QMLP-I แสดงได้ดังรูปที่ 4.12



รูปที่ 4.12 ไคอะแกรมแสดงขั้นตอนการรู้จำข้อมูลของโครงข่ายแบบ QMLP-I

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.4.3 Quantitative-classification Multi-Layer Perceptron – Type II (QMLP-II)

การรู้จำท่ามือของโครงข่ายแบบ QMLP-II จะมีกระบวนการรู้จำที่เหมือนกับการรู้จำของโครงข่ายแบบ QMLP-I ซึ่งแบ่งเป็น 2 ขั้นตอน คือ

1. การรู้จำความหมาย
2. การรู้จำขนาดหรือปริมาณ ในกรณีที่ข้อมูลที่ต้องการรู้จำเป็นท่ามือบอกขนาดหรือปริมาณ

1. การรู้จำความหมาย

เนื่องจากว่าท่ามือบอกขนาดหรือปริมาณแต่ละท่าจะมีโหนดที่ชั้นผลลัพธ์ (Output Layer) ทั้งหมด 2 โหนด ค่าทั้ง 2 โหนดจะมีวัตถุประสงค์ในการนำไปใช้ที่แตกต่างกัน ซึ่งในการรู้จำความหมายของท่ามือบอกขนาดหรือปริมาณ จะใช้โหนดที่ 1 (D_i) เพียงโหนดเดียวเท่านั้น โดยนำค่าผลลัพธ์ของโหนดที่ 1 ของท่ามือบอกขนาดหรือปริมาณแต่ละท่า ไปเปรียบเทียบกับค่าผลลัพธ์ของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ เพื่อหาว่าข้อมูลที่ต้องการรู้จำนั้นเป็นสมาชิกของท่ามือใด โดยการหาค่าที่มากที่สุด (Winner Class) ดังสมการที่ 4.10

$$x \in \omega_k \text{ if } k^{\text{th}} = \max([o_1, o_1, d_i]) \quad \text{for } i = 1, 2, \dots, v \quad (4.10)$$

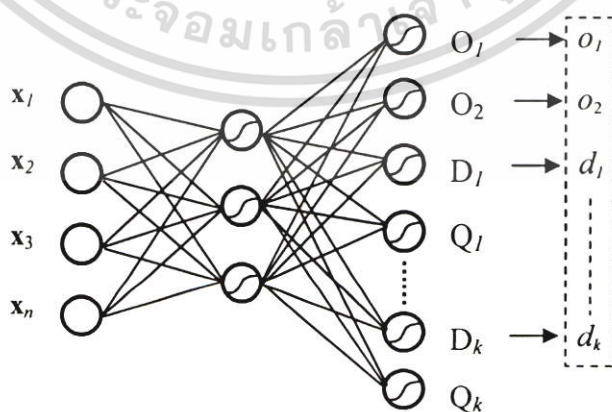
โดยที่

v คือ ท่ามือบอกขนาดทั้งหมด

o_1 คือ ค่าผลลัพธ์ของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณท่าที่ 1st

o_2 คือ ค่าผลลัพธ์ของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณท่าที่ 2nd

d_i คือ ค่าผลลัพธ์โหนดที่ 1 (D_i) ท่ามือบอกขนาดหรือปริมาณท่าที่ i^{th}

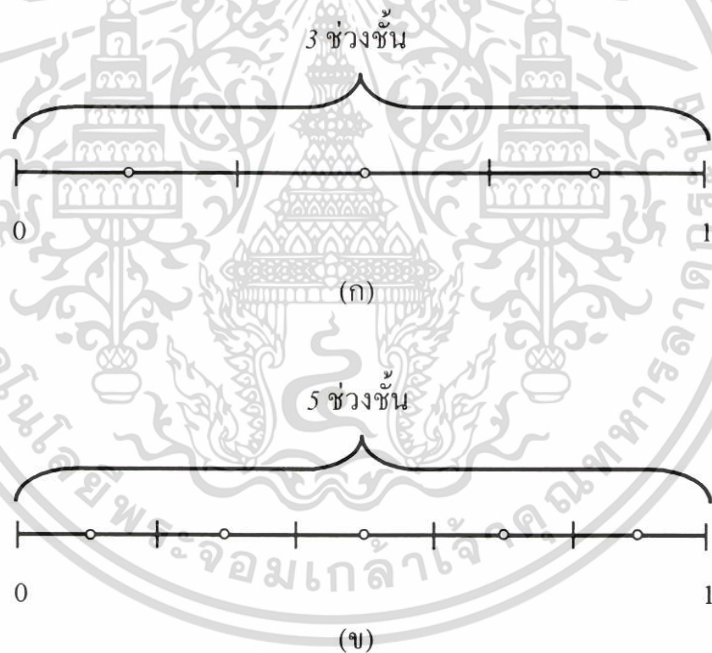


รูปที่ 4.13 แสดงการรู้จำความหมายของโครงข่ายประสาทเทียมแบบ QMLP-II

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

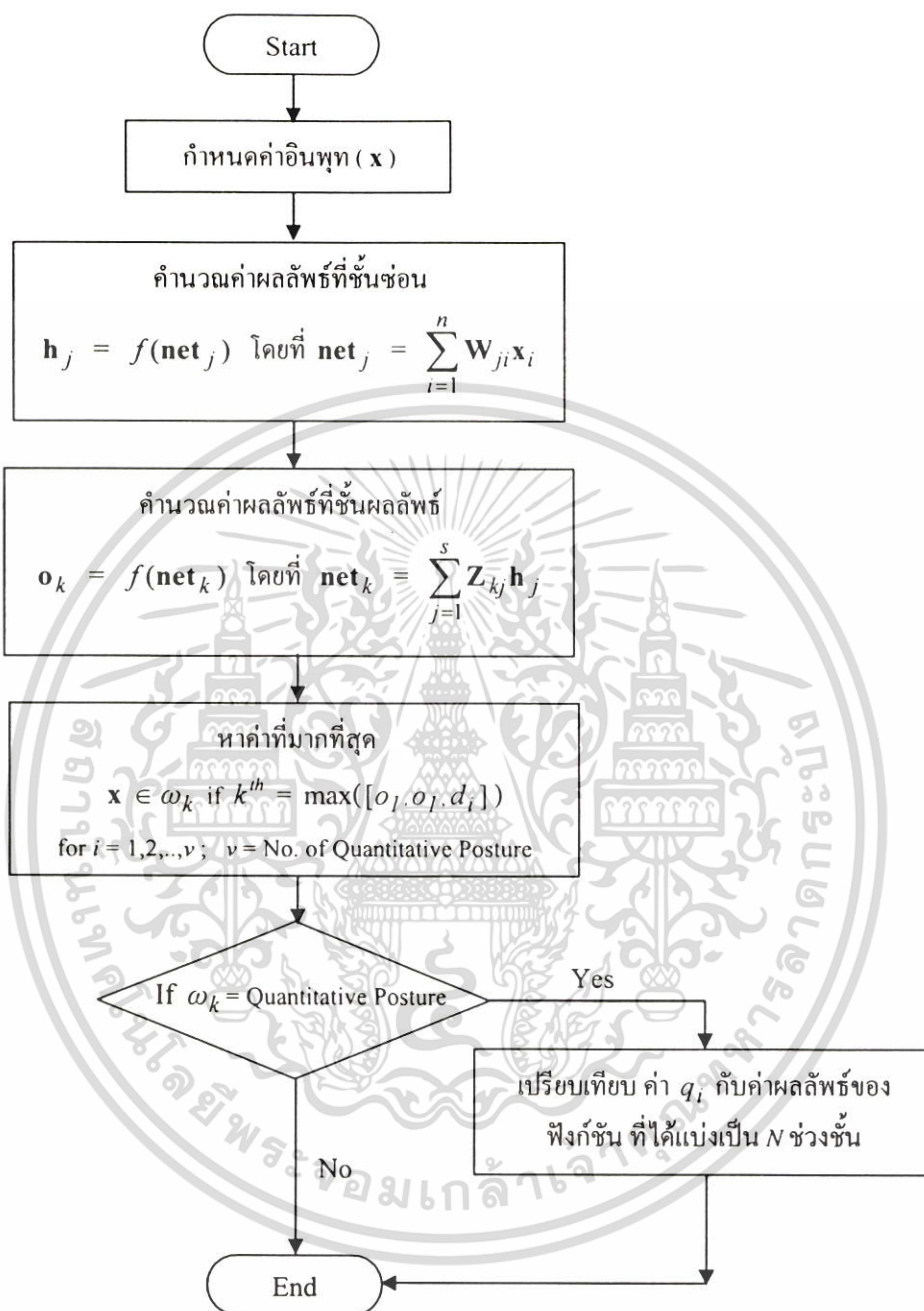
2. การรู้จำขนาดหรือปริมาณ

ในกรณีที่ทำมือบอกขนาดหรือปริมาณทำที่ i^{th} เป็นทำที่มีค่าผลลัพธ์มากที่สุด (Winner Class) จะต้องมีการระบุขนาดหรือปริมาณของทำมือนั้นด้วย โดยการนำค่าผลลัพธ์ของ โหนดที่ 2 (q_i) ของทำมือที่เป็น Winner Class มาใช้ในการระบุขนาดหรือปริมาณ การระบุขนาดของทำมือทำได้โดยการแบ่งค่าผลลัพธ์ของ Sigmoid Function ออกเป็น N ช่วงชั้น ขึ้นอยู่กับระดับขนาด หรือปริมาณที่ต้องการรู้จำ ยกตัวอย่างเช่น ถ้าหากว่าทำมือบอกขนาดหรือปริมาณแต่ละทำต้องการรู้จำขนาดหรือปริมาณทั้งหมด 3 ระดับ ก็ต้องแบ่งออกเป็น 3 ช่วงชั้น หลังจากนั้นก็นำค่าผลลัพธ์ของโหนดที่ 2 (q_i) ของทำมือที่เป็น Winner Class ไปเปรียบเทียบกับค่าผลลัพธ์อยู่ในช่วงชั้นใด แสดงว่าเวกเตอร์ข้อมูลมีขนาดหรือปริมาณระดับนั้น รูปที่ 4.14 (ก) ถ้าหากว่าทำมือบอกขนาดหรือปริมาณแต่ละทำต้องการรู้จำขนาดหรือปริมาณทั้งหมด 5 ระดับ ก็ต้องแบ่งออกเป็น 5 ช่วงชั้น แล้วยังก็นำค่าผลลัพธ์ของโหนดที่ 2 (q_i) ของทำมือที่เป็น Winner Class ไปเปรียบเทียบกับค่าผลลัพธ์ อยู่ในช่วงชั้นใด แสดงว่าเวกเตอร์ข้อมูลมีขนาดหรือปริมาณระดับนั้น รูปที่ 4.14 (ข)



รูปที่ 4.14 แสดงการรู้จำขนาดหรือปริมาณของโครงข่ายประสาทเทียมแบบ QMLP-II

ขั้นตอนการรู้จำของโครงข่ายประสาทเทียมแบบ QMLP-I แสดงได้ดังรูปที่ 4.15



รูปที่ 4.15 ไคอะแกรมแสดงขั้นตอนการรู้จำข้อมูลของโครงข่ายแบบ QMLP-II

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

อุปกรณ์และข้อมูลที่ใช้ในการทดลอง

5.1 อุปกรณ์ที่ใช้ในการทดลอง

ในส่วนนี้จะกล่าวถึงอุปกรณ์ที่ใช้ในการทดลองของงานวิจัยฉบับนี้ ซึ่งมีดังต่อไปนี้

5.1.1 ฮาร์ดแวร์ (Hardware) ประกอบด้วย

5.1.1.1 ถุงมืออิเล็กทรอนิกส์

ถุงมืออิเล็กทรอนิกส์ [13-15] เป็นอุปกรณ์ที่ใช้ในการเก็บข้อมูลของมือและตำแหน่งของมือในขณะที่กำลังทำท่ามือ ซึ่งถุงมืออิเล็กทรอนิกส์จะมีตัวต้านทาน หรือเรียกว่า เซนเซอร์ (sensor) ฝังไว้ที่ถุงมือตามข้อต่อของนิ้ว ฝ่ามือ และข้อมือ ค่าที่ได้จากเซนเซอร์เป็นตัวที่ใช้ในการกำหนดลักษณะของมือและตำแหน่งของมือ แต่ในงานวิจัยฉบับนี้ จะสนใจเฉพาะค่าของเซนเซอร์ที่เป็นค่าที่ใช้ในการระบุลักษณะของมือเท่านั้น ไม่เกี่ยวข้องกับตำแหน่งของมือ หรือการหมุนของมือ ซึ่งมีทั้งหมด 22 ค่า แต่จะใช้ค่าของเซนเซอร์เพียง 18 ค่าเท่านั้น โดยจะตัดค่าของเซนเซอร์ ณ ตำแหน่งข้อต่อที่ปลายนิ้วของนิ้วชี้ นิ้วกลาง นิ้วนาง และนิ้วก้อยออก เนื่องจากว่า ณ ตำแหน่งนี้ ถ้าหากไม่มีการจับหรือหยิบสิ่งของ ค่าเซนเซอร์ ณ ตำแหน่งเหล่านี้สามารถคำนวณได้ ค่าที่ได้จากเซนเซอร์จะมีค่าอยู่ระหว่าง 0-255 ขึ้นอยู่กับสรีระของมือของผู้สวมใส่ถุงมือ ในงานวิจัยฉบับนี้จะใช้ถุงมือข้างขวาเพียงข้างเดียวเท่านั้น ดังแสดงในรูปที่ 5.1



รูปที่ 5.1 ถุงมืออิเล็กทรอนิกส์

5.1.1.2 เครื่องคอมพิวเตอร์ เครื่องคอมพิวเตอร์ที่ใช้มีลักษณะดังนี้

- PC Intel Pentium III 800 MHz
- หน่วยความจำ 256 MB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.2 ซอฟต์แวร์ (Software) ประกอบด้วย

- Microsoft Visual C++ 6.0
- Virtual Hand Library Version 2.4
- MatLab Version 6.5
- Microsoft Access 2000
- ระบบปฏิบัติการ Microsoft Windows 2000

5.2 ข้อมูลที่ใช้ในการทดลอง

จากการศึกษาภาษามือไทย พบว่าภาษามือไทยนั้น จะประกอบไปด้วย 4 ส่วนที่สำคัญ คือ

- ท่ามือ (Hand Posture) หมายถึง การท่ามือให้เป็นท่าต่าง ๆ ขึ้นอยู่กับความหมายของคำที่ต้องการจะสื่อ
- ตำแหน่งของมือ (Position of hand or Hand location) หมายถึง ตำแหน่งของท่ามือที่กำลังทำขณะนั้น โดยอ้างอิงกับตัวของผู้กระทำท่ามือ โดยที่แต่ละตำแหน่งจะมีความหมายแตกต่างกันออกไป กล่าวคือ ท่ามือเดียวกันแต่วางอยู่คนละตำแหน่ง ก็จะมีความหมายที่แตกต่างกัน
- การเคลื่อนไหวของมือ (Hand movement) หมายถึง การเคลื่อนไหวของมือไปในทิศทางต่าง ๆ ซึ่งขึ้นอยู่กับความหมายของคำที่ต้องการจะสื่อ ในแต่ละความหมายก็จะมี การเคลื่อนไหวของมือในทิศทางที่แตกต่างกันไป
- การหมุนหรือการหักของฝ่ามือ (Orientation of Palm) หมายถึง การหันของฝ่ามือ ซึ่งอ้างอิงกับตัวผู้กระทำท่ามือ



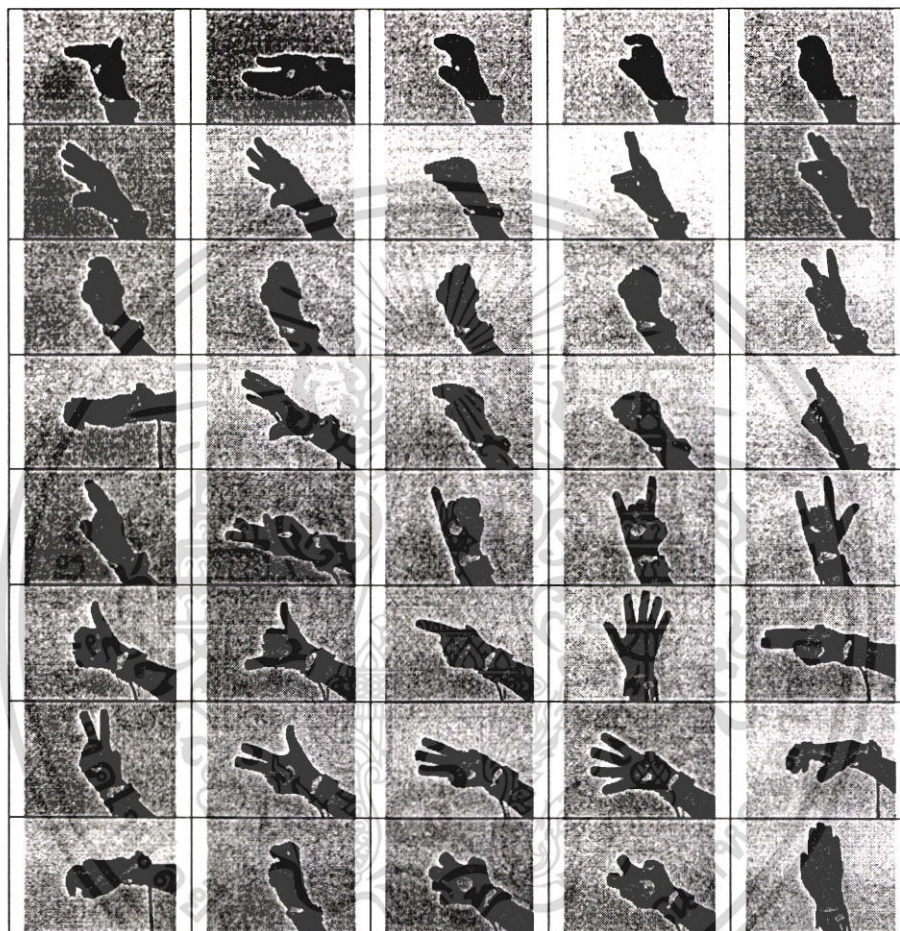
รูปที่ 5.2 ตัวอย่างท่ามือที่มีความหมายว่า “ลิม” ในระบบภาษามือไทย

ในงานวิจัยฉบับนี้ จะสนใจเฉพาะข้อมูลที่เป็นท่ามือ (Hand Posture) เท่านั้น (ไม่รวมตำแหน่งและการหมุนมุมมองของมือ) ท่ามือที่ใช้จะเป็นท่ามือพื้นฐานของภาษามือไทย ซึ่งมีลักษณะดังนี้

- เป็นท่ามือนิ่ง คือ เป็นท่ามือที่ไม่มีการเปลี่ยนท่ามือในขณะที่ทำท่ามือ (Static)
- เป็นท่ามือที่อยู่บนสมมุติฐานที่ว่า ทราบจุดเริ่มต้นและจุดสิ้นสุดของท่ามือแต่ละท่า (Isolate)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากว่าในงานวิจัยฉบับนี้ ประกอบด้วย 2 ส่วน คือ ส่วนที่หนึ่ง จะเป็นการเปรียบเทียบวิธีการในการรู้จำ ข้อมูลที่ใช้จะเป็นท่ามือพื้นฐานทั้งหมด 40 ท่า ดังรูปที่ 5.3 และในส่วนที่สองของงานวิจัย คือ การพัฒนาโครงข่ายประสาทเทียมในการรู้จำท่ามือบอกขนาดหรือปริมาณ ข้อมูลที่ใช้คือ ท่ามือพื้นฐานที่เป็นท่าบอกขนาดหรือปริมาณทั้งหมด 3 ท่า ท่าละ 5 ระดับ ดังรูปที่ 5.4 โดยจะทดลองรวมกับรูปมือพื้นฐานทั้ง 40 ท่าในส่วนที่หนึ่ง



รูปที่ 5.3 ตัวอย่างท่ามือพื้นฐานของภาษามือไทย



(ก) ท่ามือบอกขนาดหรือปริมาณที่มีความหมายว่า “ความหนาของวัตถุ”

รูปที่ 5.4 ตัวอย่างท่ามือบอกขนาดหรือปริมาณ (Quantitative Posture)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ข) ทำมือบอกขนาดหรือปริมาณที่มีความหมายว่า “ความสว่างของแสง”



(ค) ทำมือบอกขนาดหรือปริมาณที่มีความหมายว่า “ขนาดของวัดอุทรทรงกลม”

รูปที่ 5.4 (ต่อ) ตัวอย่างท่ามือบอกขนาดหรือปริมาณ (Quantitative Posture)

ในงานวิจัยฉบับนี้ ใช้ผู้ทดลองทั้งหมด 10 คน ซึ่งแต่ละคนจะมีลักษณะสรีระของมือที่แตกต่างกันไป การเก็บข้อมูลจะให้ผู้ทดลองสวมถุงมืออิเล็กทรอนิกส์ที่เชื่อมต่อกับเครื่องคอมพิวเตอร์ ในการเก็บข้อมูลจะแยกออกเป็น 2 ส่วน คือ ในส่วนที่หนึ่งของงานวิจัย จะเก็บข้อมูลทั้งหมด 40 ท่า ทำท่าละ 40 ครั้ง (ชุดข้อมูล) ดังนั้น ผู้ทดลองแต่ละคนจะเก็บข้อมูลในส่วนที่หนึ่งทั้งหมด 1,600 ครั้ง และในส่วนที่สองจะเก็บข้อมูลของท่ามือบอกขนาดหรือปริมาณทั้งหมด 3 ท่า ทำท่าละ 5 ระดับ ระดับละ 40 ครั้ง ดังนั้น ผู้ทดลองแต่ละคนต้องเก็บข้อมูลในส่วนที่สองทั้งหมด 600 ครั้ง รวมทั้งสองส่วน ผู้ทดลองแต่ละคนจะต้องเก็บข้อมูลทั้งหมด 2,200 ครั้ง

บทที่ 6

วิธีการทดลองและผลการทดลอง

6.1 การเปรียบเทียบวิธีการสำหรับรู้จำท่ามือ

6.1.1 วิธีการทดลอง

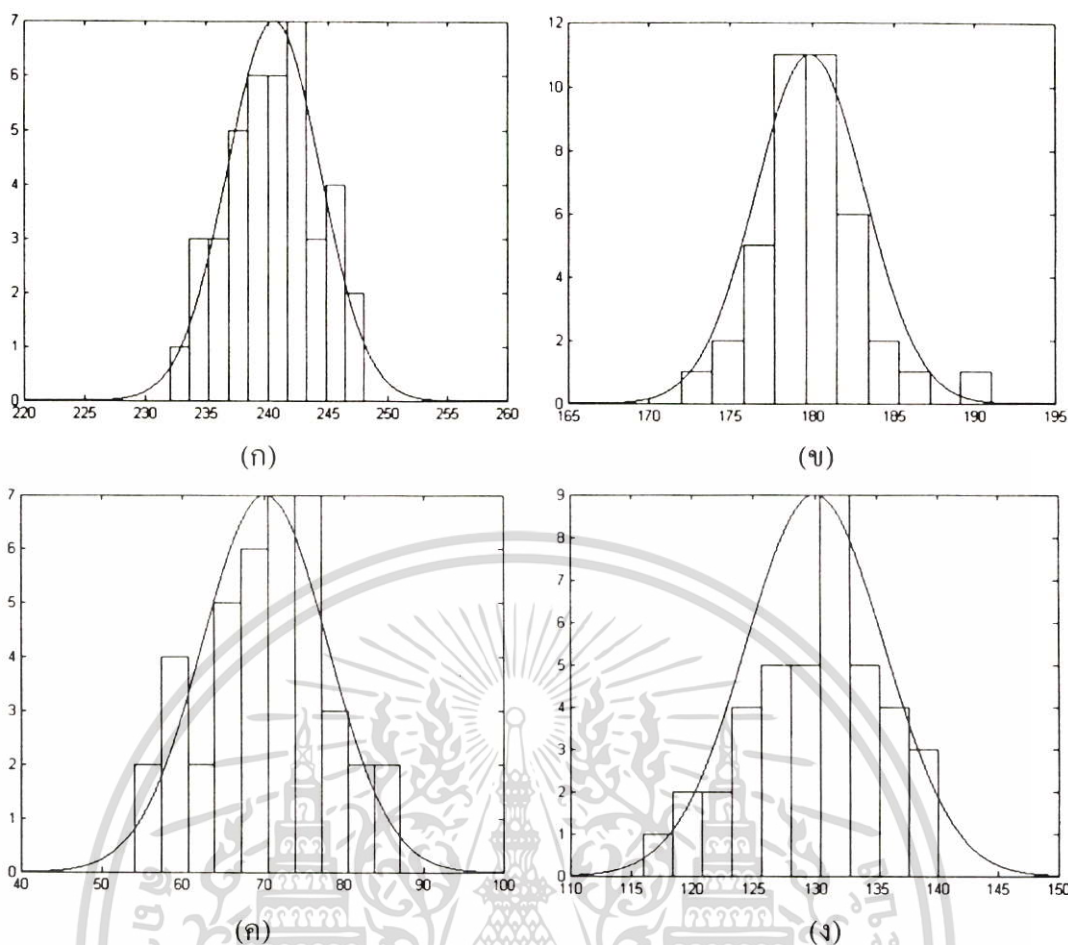
การทดลองในส่วนที่หนึ่ง เป็นการเปรียบเทียบวิธีการสำหรับรู้จำท่ามือพื้นฐานในภาษาไทย วิธีการที่จะนำมาเปรียบเทียบ คือ โครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron (MLP) และ Radial Basis Function (RBF) และ Bayesian Estimator ซึ่งจะเปรียบเทียบในประเด็นของความถูกต้อง และความซับซ้อนในการคำนวณขณะแปลความหมาย โดยที่การเปรียบเทียบความซับซ้อนในการคำนวณขณะแปลความหมายจะพิจารณาจากจุดที่วิธีการที่เปรียบเทียบทั้ง 3 วิธี มีค่าความถูกต้องที่ใกล้เคียงกัน

ในการทดลอง มีจำนวนผู้ทดลองทั้งหมด 10 คน แต่ละคนจะเก็บข้อมูลของท่ามือพื้นฐานเพื่อใช้ในการทดลองส่วนที่หนึ่งของงานวิจัยทั้งหมด 40 ท่ามือ ท่ามือละ 40 ชุดข้อมูล การทดลองจะแบ่งผู้ทดลองออกเป็น 2 กลุ่ม กลุ่มละ 5 คน โดยข้อมูลของผู้ทดลองแต่ละกลุ่มจะนำไปใช้ในการทดลอง 2 กรณี คือ

1. Signer Dependence คือ เป็นการทดลองโดยใช้ข้อมูลของผู้ทดลองแต่ละคนในกลุ่มผู้ทดลองกลุ่มแรก มาทำการเรียนรู้และทดสอบแยกกันทีละคน กล่าวคือ นำข้อมูลในส่วนที่หนึ่งของแต่ละท่า ของผู้ทดลองคนที่ 1 มาใช้ในการเรียนรู้ และนำข้อมูลส่วนที่สองของผู้ทดลองคนที่ 1 มาใช้ในการทดสอบ หลังจากนั้นก็ทำการทดลองเช่นเดียวกับผู้ทดลองคนที่ 1 กับผู้ทดลองที่เหลือในกลุ่มผู้ทดลองกลุ่มแรก แล้วหาค่าเฉลี่ยของผลการทดลองของผู้ทดลองทุกคนในกลุ่มที่หนึ่ง

2. Signer Independence คือ เป็นการทดลองโดยใช้ข้อมูลบางส่วนของกลุ่มผู้ทดลองกลุ่มแรกมาใช้ในการเรียนรู้ของวิธีการทั้ง 3 วิธี และนำข้อมูลบางส่วนของกลุ่มผู้ทดลองกลุ่มที่สองมาใช้ในการทดสอบ กล่าวคือ นำข้อมูลส่วนที่หนึ่งของกลุ่มผู้ทดลองกลุ่มแรก (กลุ่มผู้ทดลองในกรณีของ Signer Dependence) มาใช้ในการเรียนรู้ และนำข้อมูลบางส่วนของกลุ่มผู้ทดลองกลุ่มที่สองมาใช้ในการทดสอบ

จากการวิเคราะห์ลักษณะของข้อมูลที่ใช้ในการทดลอง พบว่าข้อมูลมีการกระจายใกล้เคียงกับการกระจายแบบปกติ ดังแสดงในรูปที่ 6.1 ดังนั้น วิธีการแบบ Bayesian Estimator จึงใช้สมการของการกระจายแบบปกติในการจำลอง



รูปที่ 6.1 Histogram และเส้นโค้งปกติของข้อมูลที่ใช้ในการทดลอง

6.1.2 ผลการทดลอง

ในส่วนของผลการทดลอง จะแบ่งออกเป็น 2 ส่วน คือ ส่วนของการเปรียบเทียบความถูกต้อง และส่วนของการเปรียบเทียบความซับซ้อนในการคำนวณขณะแปลความหมาย ซึ่งทั้งสองส่วนจะทดลองเปรียบเทียบใน 2 กรณี คือ Signer Dependence และ Signer Independence

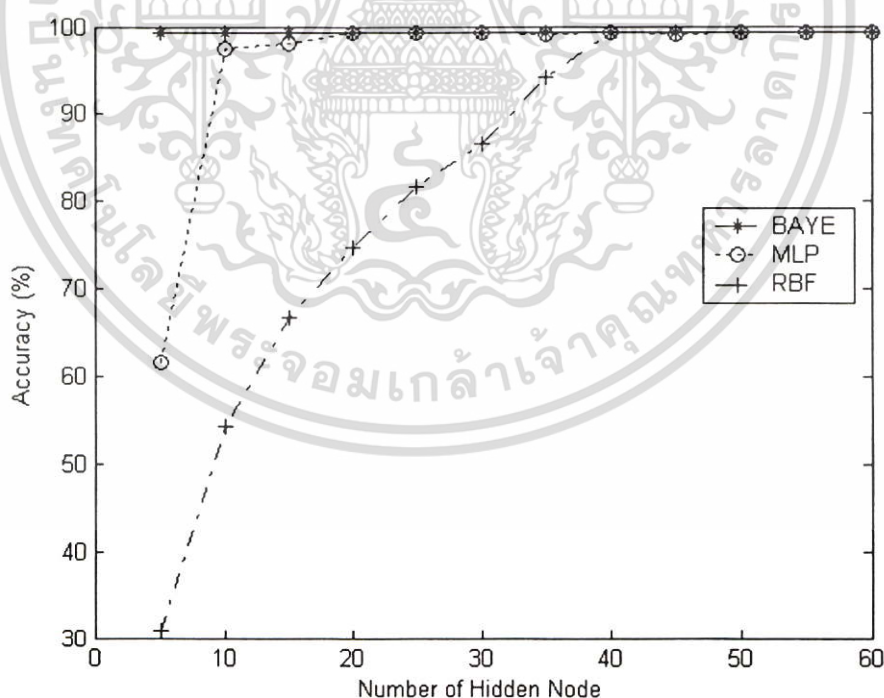
6.1.2.1 การเปรียบเทียบความถูกต้อง

การเปรียบเทียบความถูกต้องระหว่างโครงข่ายประสาทเทียมทั้ง 2 แบบ กับวิธีการแบบ Bayesian Estimator จะทดลองโดยการใช้จำนวนโนนคในชั้นซ่อน (Hidden Layer) ที่แตกต่างกัน เพื่อหาจำนวนโนนคที่ทำให้ค่าความถูกต้องของโครงข่ายประสาทเทียมทั้ง 2 แบบ มีค่าใกล้เคียงกับวิธีการแบบ Bayesian Estimator

ตารางที่ 6.1 แสดงค่าความถูกต้องในกรณีของ Signer Dependence

Method	Accuracy (%)											
	Number of Hidden Node											
	5	10	15	20	25	30	35	40	45	50	55	60
BAYE	99.30	99.30	99.30	99.30	99.30	99.30	99.30	99.30	99.30	99.30	99.30	99.30
MLP	61.58	97.35	98.08	99.25	99.23	99.20	99.18	99.20	99.18	99.20	99.23	99.20
RBF	30.83	54.18	66.65	74.75	81.60	86.55	94.15	99.23	99.23	99.20	99.25	99.20

ตารางที่ 6.1 แสดงค่าความถูกต้องของวิธีการทั้ง 3 วิธี ในกรณีของ Signer Dependence ผลการทดลองพบว่า วิธีการที่เปรียบเทียบทั้ง 3 วิธี มีค่าความถูกต้องที่ใกล้เคียงกัน จากตารางที่ 6.1 พบว่า ค่าความถูกต้องของโครงข่ายประสาทเทียมทั้ง 2 แบบมีแนวโน้มเพิ่มขึ้น เมื่อจำนวนโหนดเพิ่มขึ้น จนกระทั่งถึงจุดที่ค่าความถูกต้องเริ่มคงที่ ในส่วนของโครงข่ายประสาทเทียมแบบ MLP จำนวนโหนดที่ทำให้ค่าความถูกต้องใกล้เคียงกับวิธีการ Bayesian Estimator คือ 20 โหนด ซึ่งค่าความถูกต้องต่างกันเพียง 0.05% ส่วนโครงข่ายประสาทเทียมแบบ RBF จำนวนโหนดที่ทำให้ค่าความถูกต้องใกล้เคียงกับวิธีการ Bayesian Estimator คือ 40 โหนด ซึ่งค่าความถูกต้องต่างกันเพียง 0.07% ตารางที่ 6.1 สามารถแสดงได้ดังรูปที่ 6.2



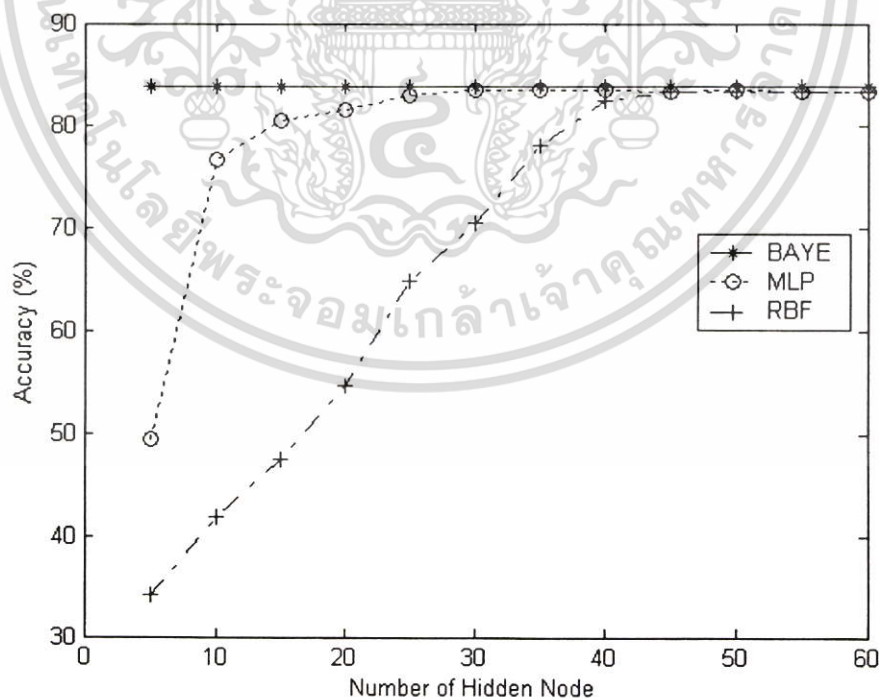
รูปที่ 6.2 แสดงค่าความถูกต้องของวิธีการที่ใช้ในการเปรียบเทียบในกรณี Signer Dependence

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6.2 แสดงค่าความถูกต้องในกรณีของ Signer Independence

Method	Accuracy (%)											
	Number of Hidden Node											
	5	10	15	20	25	30	35	40	45	50	55	60
BAYE	83.75	83.75	83.75	83.75	83.75	83.75	83.75	83.75	83.75	83.75	83.75	83.75
MLP	49.33	76.58	80.43	81.48	82.98	83.40	83.35	83.38	83.33	83.35	83.30	83.28
RBF	34.15	41.90	47.38	54.63	64.80	70.43	78.05	82.35	83.33	83.30	83.33	83.28

ตารางที่ 6.2 แสดงค่าความถูกต้องของวิธีการทั้ง 3 วิธี กรณีของ Signer Independence จากตารางพบว่า ผลการทดลองมีความคล้ายคลึงกับผลการทดลองในกรณี Signer Dependence กล่าวคือ วิธีการที่เปรียบเทียบทั้ง 3 วิธี มีค่าความถูกต้องที่ใกล้เคียงกัน จากตารางที่ 6.2 พบว่า ค่าความถูกต้องของโครงข่ายประสาทเทียมทั้ง 2 แบบมีแนวโน้มเพิ่มขึ้น เมื่อจำนวนโหนดเพิ่มขึ้น จนกระทั่งถึงจุดที่ค่าความถูกต้องเริ่มคงที่ ในส่วนของโครงข่ายประสาทเทียมแบบ MLP จำนวนโหนดที่ทำให้ค่าความถูกต้องใกล้เคียงกับวิธีการ Bayesian Estimator คือ 30 โหนด ซึ่งค่าความถูกต้องต่างกันเพียง 0.35% ส่วนโครงข่ายประสาทเทียมแบบ RBF จำนวนโหนดที่ทำให้ค่าความถูกต้องใกล้เคียงกับวิธีการ Bayesian Estimator คือ 45 โหนด ซึ่งค่าความถูกต้องต่างกันเพียง 0.42% ตารางที่ 6.2 สามารถแสดงได้ดังรูปที่ 6.3



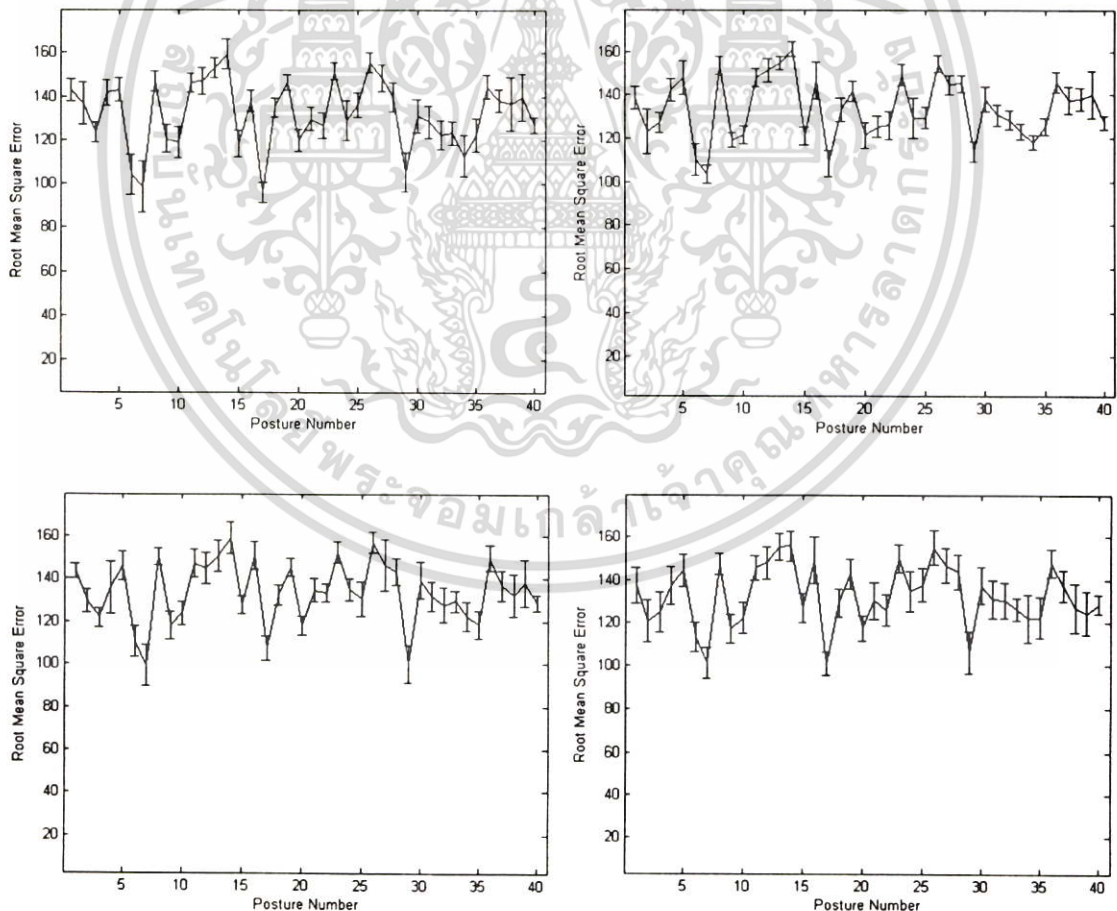
รูปที่ 6.3 แสดงค่าความถูกต้องของวิธีการที่ใช้ในการเปรียบเทียบในกรณี Signer Independence

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการทดลองในกรณี Signer Dependence พบว่าวิธีการแบบ Bayesian Estimator และโครงข่ายประสาทเทียมแบบ MLP และ RBF มีค่าความถูกต้องที่สูงมาก (99.30%, 99.25% และ 99.23% ตามลำดับ) ซึ่งจากการวิเคราะห์ข้อมูลโดยการหาค่า Root Mean Square Error (RMSE) ดังสมการที่ 6.1 พบว่า ข้อมูลในแต่ละท่ามือนั้นมีค่า RMSE ที่น้อยมาก แสดงว่าข้อมูลมีความผิดพลาดในการทำท่ามือเมื่อเทียบกับค่าเฉลี่ยของแต่ละท่ามือน้อยมาก ดังแสดงในรูปที่ 6.4

$$\text{RMSE} = \sqrt{\frac{1}{p} \sum_{j=1}^p \frac{1}{r} \sum_{i=1}^r (v_{i,j} - \bar{v}_i)^2} \quad (6.1)$$

โดยที่ $v_{i,j}$ คือ ค่าของเซนเซอร์ตัวที่ i^{th} ของชุดข้อมูล j^{th}
 \bar{v}_i คือ ค่าเฉลี่ยของเซนเซอร์ตัวที่ i^{th} ของท่ามือแต่ละท่า
 r คือ จำนวนเซนเซอร์ทั้งหมด
 p คือ จำนวนของชุดข้อมูลทั้งหมดในแต่ละท่ามือ



รูปที่ 6.4 แสดงค่า RMSE ของท่ามือแต่ละท่า กรณี Signer Dependence

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.1.2.2 การเปรียบเทียบความซับซ้อนในการคำนวณขณะแปลความหมาย

การเปรียบเทียบความซับซ้อนในการคำนวณขณะแปลความหมายระหว่างโครงข่ายประสาทเทียมทั้ง 2 แบบกับวิธีการแบบ Bayesian Estimator จะพิจารณาจากจุดที่วิธีทั้ง 3 วิธี มีความถูกต้องที่ใกล้เคียงกัน โดยใช้ผลการทดลองในส่วนของเปรียบเทียบความถูกต้องเป็นหลัก ซึ่งจะแบ่งเป็น 2 กรณี คือ Signer Dependence และ Signer Independence ซึ่งสมการที่ใช้ในการคำนวณความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการทั้ง 3 วิธี สรุปได้ดังนี้

กำหนดให้

n คือ ขนาดมิติ (Dimension) ของเวกเตอร์ข้อมูล (Input Vector) ซึ่งก็คือจำนวนของเซนเซอร์ ในที่นี้มีค่าเท่ากับ 18

s คือ จำนวน โหนดในชั้นซ่อน

m คือ จำนวนของท่ามือ (Classes) ทั้งหมด ในที่นี้มีค่าเท่ากับ 40

- โครงข่ายประสาทเทียมแบบ MLP

$$\text{การบวก} = (n-1)s + s + (s-1)m + m \quad (6.2)$$

$$\text{การคูณ} = (n+m)s \quad (6.3)$$

$$\text{การหาร} = s + m \quad (6.4)$$

$$\text{ยกกำลัง} = s + m \quad (6.5)$$

- โครงข่ายประสาทเทียมแบบ RBF

$$\text{การบวก} = (n-1)s + (s-1)m \quad (6.6)$$

$$\text{การลบ} = sn \quad (6.7)$$

$$\text{การคูณ} = ms \quad (6.8)$$

$$\text{การหาร} = s \quad (6.9)$$

$$\text{ยกกำลัง} = (n+1)s \quad (6.10)$$

- วิธีการแบบ Bayesian Estimator

$$\text{การบวก} = [(n-1)n + n]m \quad (6.11)$$

$$\text{การลบ} = nm \quad (6.12)$$

$$\text{การคูณ} = (n^2 + n)m \quad (6.13)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางที่ 6.1 ในกรณีของ Signer Dependence ในส่วนของโครงข่ายประสาทเทียมแบบ MLP จำนวนโหนดที่ทำให้ค่าความถูกต้องของโครงข่ายมีค่าใกล้เคียงกับ Bayesian Estimator คือ 20 โหนด และในส่วนของโครงข่ายประสาทเทียมแบบ RBF จำนวนโหนดที่ทำให้ค่าความถูกต้องของโครงข่ายมีค่าใกล้เคียงกับ Bayesian Estimator คือ 40 โหนด ดังนั้น ความซับซ้อนในการคำนวณขณะแปลความหมายของโครงข่ายประสาทเทียมแบบ MLP และ RBF จะใช้จำนวนโหนดเท่ากับ 20 และ 40 โหนด ตามลำดับ ซึ่งผลการคำนวณแสดงในตารางที่ 6.3

ตารางที่ 6.3 แสดงความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการที่ใช้ในการเปรียบเทียบในกรณี Signer Dependence

วิธีการ	ความซับซ้อนในการคำนวณ (ครั้ง)					
	บวก	ลบ	คูณ	หาร	ยกกำลัง	รวม
BAYE	12,960	720	13,680	0	0	27,360
MLP	1,160	0	1,160	60	60	2,440
RBF	2,240	720	1,600	40	760	5,360

จากตารางที่ 6.3 ผลการคำนวณพบว่า ในกรณีของ Signer Dependence ที่ระดับความถูกต้องที่ใกล้เคียงกันของวิธีการทั้ง 3 วิธี โครงข่ายประสาทเทียมแบบ MLP จะมีความซับซ้อนในการคำนวณขณะแปลความหมายน้อยที่สุด รองลงมาเป็นโครงข่ายประสาทเทียมแบบ RBF ส่วน Bayesian Estimator จะมีความซับซ้อนในการคำนวณขณะแปลความหมายมากที่สุด โดยเฉพาะการบวกและการคูณซึ่งวิธี Bayesian Estimator จะมีการคำนวณที่มากกว่าโครงข่ายประสาทเทียมแบบ MLP ประมาณ 11.2 และ 11.8 เท่า ตามลำดับ และมากกว่าโครงข่ายประสาทเทียมแบบ RBF ประมาณ 5.8 และ 8.5 เท่า ตามลำดับ

ส่วนความซับซ้อนในการคำนวณขณะแปลความหมายกรณีของ Signer Independence พิจารณาจากตารางที่ 6.2 จำนวนโหนดที่ทำให้ค่าความถูกต้องของโครงข่ายประสาทเทียมแบบ MLP และ RBF มีค่าใกล้เคียงกับ Bayesian Estimator คือ 30 และ 45 โหนด ตามลำดับ ดังนั้น ความซับซ้อนในการคำนวณขณะแปลความหมายของโครงข่ายประสาทเทียมแบบ MLP และ RBF จะใช้จำนวนโหนดเท่ากับ 30 และ 45 โหนด ตามลำดับ ซึ่งผลการคำนวณแสดงในตารางที่ 6.4

ตารางที่ 6.4 แสดงความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการที่ใช้ในการ
เปรียบเทียบในกรณี Signer Independence

วิธีการ	ความซับซ้อนในการคำนวณ (ครั้ง)					
	บวก	ลบ	คูณ	หาร	ยกกำลัง	รวม
BAYE	12,960	720	13,680	0	0	27,360
MLP	1,740	0	1,740	70	70	3,620
RBF	2,525	810	1,800	45	855	6,035

จากตารางที่ 6.4 ผลการคำนวณพบว่า ในกรณีของ Signer Independence ที่ระดับความถูกต้องที่ใกล้เคียงกันของวิธีการทั้ง 3 วิธี ซึ่งโครงข่ายประสาทเทียมแบบ MLP ยังมีความซับซ้อนในการคำนวณขณะแปลความหมายน้อยที่สุด ซึ่ง Bayesian Estimator จะมีความซับซ้อนในการคำนวณขณะแปลความหมายมากที่สุด โดยเฉพาะการบวกและการคูณซึ่งวิธี Bayesian Estimator จะมีการคำนวณที่มากกว่าโครงข่ายประสาทเทียมแบบ MLP ประมาณ 7.5 และ 7.8 เท่า ตามลำดับ และมากกว่าโครงข่ายประสาทเทียมแบบ RBF ประมาณ 5 และ 7.6 เท่า ตามลำดับ

6.1.3 สรุปผลการทดลอง

จากผลการทดลอง ในส่วนของการเปรียบเทียบความถูกต้อง สรุปได้ว่า ถ้าจำนวนโหนดในชั้นซ่อนมีจำนวนเพียงพอ โครงข่ายประสาทเทียมจะมีค่าความถูกต้องใกล้เคียงกับวิธีการ Bayesian Estimator ในส่วนของความซับซ้อนของการคำนวณขณะแปลความหมาย สรุปได้ว่า ที่ระดับความถูกต้องที่ใกล้เคียงกัน โครงข่ายประสาทเทียมแบบ MLP จะมีความซับซ้อนในการคำนวณขณะแปลความหมายน้อยที่สุด

6.2 การพัฒนาโครงข่ายประสาทเทียมสำหรับรู้จำท่ามือบอกขนาดหรือปริมาณ

6.2.1 วิธีการทดลอง

การทดลองในส่วนที่สองของงานวิจัย เป็นการเปรียบเทียบค่าความถูกต้องในการรู้จำท่ามือบอกขนาดหรือปริมาณ (Quantitative Posture) ที่พบในภาษามือไทย ระหว่างการใช้โครงข่ายประสาทเทียมแบบ Standard Multi-Layer Perceptron (SMLP) กับโครงข่ายที่ได้พัฒนาสำหรับรู้จำท่ามือบอกขนาดหรือปริมาณ ซึ่งเรียกโครงข่ายที่พัฒนานี้ว่า Quantitative-classification Multi-Layer Perceptron (QMLP) ในการรู้จำท่ามือบอกขนาดหรือปริมาณ

ข้อมูลที่ใช้ในการทดลองในส่วนนี้ จะเป็นข้อมูลของท่ามือที่เป็นท่าบอกขนาดหรือปริมาณ (Quantitative Posture) ทั้งหมด 3 ท่า ในแต่ละท่าจะมีจำนวนระดับขนาดหรือปริมาณ (Quantitative Level) ทั้งหมด 5 ระดับ โดยจะทดลองร่วมกับข้อมูลของท่ามือพื้นฐานที่ใช้ในการทดลองในส่วนของการเปรียบเทียบวิธีการสำหรับรู้จำทั้ง 40 ท่า ซึ่งในส่วนนี้จะเรียกท่ามือทั้ง 40 ท่านี้ว่า ท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (Non-Quantitative Posture)

วิธีการทดลองในส่วนนี้ จะทดลองเปรียบเทียบประสิทธิภาพในการรู้จำท่ามือพื้นฐานทั้งท่ามือบอกขนาดหรือปริมาณและท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (Non-Quantitative Posture) ซึ่งโครงข่ายประสาทแบบ Standard Multi-Layer Perceptron (SMLP) และโครงข่ายประสาทเทียมแบบ Quantitative-classification Multi-Layer Perceptron (QMLP) ทั้ง 2 แบบ จะมีความต้องการข้อมูลของท่ามือบอกขนาดหรือปริมาณในการเรียนรู้ที่แตกต่างกัน กล่าวคือ โครงข่ายประสาทเทียมแบบ SMLP ต้องการข้อมูลระดับขนาดหรือปริมาณทุกระดับที่ต้องการรู้จำในการเรียนรู้ ยกตัวอย่างคือ ในท่ามือบอกขนาดหรือปริมาณแต่ละท่า ต้องใช้ข้อมูลระดับขนาดหรือปริมาณทั้งหมด 3 ระดับในการเรียนรู้ของโครงข่าย ในขณะที่ต้องการรู้จำระดับขนาดหรือปริมาณทั้งหมด 3 ระดับ และใช้ข้อมูลระดับขนาดหรือปริมาณ 5 ระดับ ในการเรียนรู้ ในขณะที่ต้องการรู้จำระดับขนาดหรือปริมาณทั้งหมด 5 ระดับ แต่ในกรณีของโครงข่ายแบบ QMLP จะใช้ข้อมูลระดับขนาดหรือปริมาณท่ามือละ 3 ระดับเท่านั้นในการเรียนรู้ แต่สามารถที่จะรู้จำระดับขนาดหรือปริมาณได้ทั้ง 3 ระดับและ 5 ระดับ การทดลองจะแบ่งเป็น 2 กรณี คือ Signer Dependence และ Signer Independence

6.2.2 ผลการทดลอง

ผลการทดลองแสดงในตารางที่ 6.5 และ 6.6

ตารางที่ 6.5 เปรียบเทียบค่าความถูกต้องของการใช้โครงข่ายประสาทเทียมแบบ SMLP และโครงข่ายประสาทเทียมแบบ QMLP ในการรู้จำท่ามือบอกขนาดหรือปริมาณในกรณีของ Signer Dependence

Method	Accuracy (%)					
	3 Levels			5 Levels		
	Non-QP Meaning	QP Meaning	QP Level	Non-QP Meaning	QP Meaning	QP Level
SMLP	99.41	99.65	99.92	99.43	99.65	97.38
QMLP-I	99.28	99.83	99.67	99.28	99.79	96.09
QMLP-II	99.20	99.87	98.67	99.20	99.75	94.64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6.5 เป็นการเปรียบเทียบความถูกต้องในการรู้จำท่ามือทั้ง 2 แบบ ในกรณีของ Signer Dependence ของโครงข่ายทั้ง 3 แบบ จากตารางพบว่า

- กรณีที่ต้องการรู้จำระดับขนาดหรือปริมาณ 3 ระดับ (3 Levels)

โครงข่ายประสาทเทียมแบบ SMLP, QMLP-I และ QMLP-II ในส่วนของการรู้จำความหมายของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (Non-Quantitative Posture) จากตารางคือ Non-QP Meaning จะมีค่าความถูกต้องที่ใกล้เคียงกัน คือ 99.41%, 99.28% และ 99.20% ตามลำดับ ส่วนการรู้จำความหมายของท่ามือที่เป็นท่าบอกขนาดหรือปริมาณ จากตารางคือ QP Meaning ก็มีค่าความถูกต้องที่ใกล้เคียงกัน คือ 99.65%, 99.83% และ 99.87% ตามลำดับ และในส่วนของการรู้จำระดับขนาดหรือปริมาณของท่ามือบอกขนาดหรือปริมาณ จากตารางคือ QP Level พบว่า โครงข่ายประสาทเทียมแบบ QMLP-I และ QMLP-II มีค่าความถูกต้องต่ำกว่าโครงข่ายประสาทเทียมแบบ SMLP ประมาณ 0.25% และ 1.25% ตามลำดับ

- กรณีที่ต้องการรู้จำระดับขนาดหรือปริมาณ 5 ระดับ (5 Levels)

โครงข่ายประสาทเทียมแบบ SMLP, QMLP-I และ QMLP-II ในส่วนของการรู้จำความหมายของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (Non-Quantitative Posture) จากตารางคือ Non-QP Meaning จะมีค่าความถูกต้องที่ใกล้เคียงกัน คือ 99.43%, 99.28% และ 99.20% ตามลำดับ ส่วนการรู้จำความหมายของท่ามือที่เป็นท่าบอกขนาดหรือปริมาณ จากตารางคือ QP Meaning ก็มีค่าความถูกต้องที่ใกล้เคียงกัน คือ 99.65%, 99.79% และ 99.75% ตามลำดับ และในส่วนของการรู้จำระดับขนาดหรือปริมาณของท่ามือบอกขนาดหรือปริมาณ จากตารางคือ QP Level พบว่า โครงข่ายประสาทเทียมแบบ QMLP-I และ QMLP-II มีค่าความถูกต้องต่ำกว่าโครงข่ายประสาทเทียมแบบ SMLP ประมาณ 1.29% และ 2.74% ตามลำดับ

ตารางที่ 6.6 เปรียบเทียบค่าความถูกต้องของการใช้โครงข่ายประสาทเทียมแบบ SMLP และโครงข่ายประสาทเทียมแบบ QMLP ในการรู้จำท่ามือบอกขนาดหรือปริมาณในกรณีของ Signer Independence

Method	Accuracy (%)					
	3 Levels			5 Levels		
	Non-QP Meaning	QP Meaning	QP Level	Non-QP Meaning	QP Meaning	QP Level
SMLP	82.05	94.27	97.78	81.65	94.70	93.09
QMLP-I	81.70	94.44	97.07	81.70	94.87	92.45
QMLP-II	81.67	94.15	94.88	81.67	94.57	90.74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 6.6 เป็นการเปรียบเทียบความถูกต้องในการรู้จำท่ามือทั้ง 2 แบบ ในกรณีของ Signer Independence ของโครงข่ายทั้ง 3 แบบ จากตารางพบว่า

- กรณีที่ต้องการรู้จำระดับขนาดหรือปริมาณ 3 ระดับ (3 Levels)

โครงข่ายประสาทเทียมแบบ SMLP, QMLP-I และ QMLP-II ในส่วนของการรู้จำความหมายของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (Non-Quantitative Posture) จากตารางคือ Non-QP Meaning จะมีค่าความถูกต้องที่ใกล้เคียงกัน คือ 82.05%, 81.70% และ 81.67% ตามลำดับ ส่วนการรู้จำความหมายของท่ามือที่เป็นท่าบอกขนาดหรือปริมาณ จากตารางคือ QP Meaning ก็มีค่าความถูกต้องที่ใกล้เคียงกัน คือ 94.27%, 94.44% และ 94.15% ตามลำดับ และในส่วนของการรู้จำระดับขนาดหรือปริมาณของท่ามือบอกขนาดหรือปริมาณ จากตารางคือ QP Level พบว่า โครงข่ายประสาทเทียมแบบ QMLP-I และ QMLP-II มีค่าความถูกต้องต่ำกว่าโครงข่ายประสาทเทียมแบบ SMLP ประมาณ 0.72% และ 2.90% ตามลำดับ

- กรณีที่ต้องการรู้จำระดับขนาดหรือปริมาณ 5 ระดับ (5 Levels)

โครงข่ายประสาทเทียมแบบ SMLP, QMLP-I และ QMLP-II ในส่วนของการรู้จำความหมายของท่ามือที่ไม่ใช่ท่าบอกขนาดหรือปริมาณ (Non-Quantitative Posture) จากตารางคือ Non-QP Meaning จะมีค่าความถูกต้องที่ใกล้เคียงกัน คือ 81.65%, 81.70% และ 81.67% ตามลำดับ ส่วนการรู้จำความหมายของท่ามือที่เป็นท่าบอกขนาดหรือปริมาณ จากตารางคือ QP meaning ก็มีค่าความถูกต้องที่ใกล้เคียงกัน คือ 94.70%, 94.87% และ 94.57% ตามลำดับ และในส่วนของการรู้จำระดับขนาดหรือปริมาณของท่ามือบอกขนาดหรือปริมาณ จากตารางคือ QP Level พบว่า โครงข่ายประสาทเทียมแบบ QMLP-I และ QMLP-II มีค่าความถูกต้องต่ำกว่าโครงข่ายประสาทเทียมแบบ SMLP ประมาณ 0.64% และ 2.35% ตามลำดับ

ความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการที่นำมาเปรียบเทียบ กรณีที่ทำมือบอกขนาดหรือปริมาณต้องการรู้จำขนาดหรือปริมาณทั้งหมด 5 ระดับ แสดงดังตารางที่ 6.7

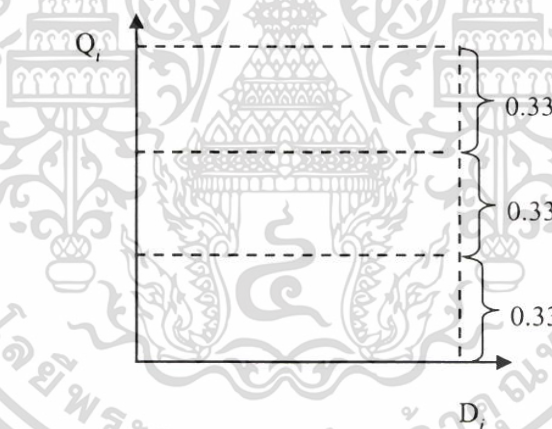
ตารางที่ 6.7 แสดงความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการที่ใช้ในการเปรียบเทียบ กรณีที่ทำมือบอกขนาดหรือปริมาณแต่ละท่าต้องการรู้จำขนาดหรือปริมาณ 5 ระดับ

วิธีการ	ความซับซ้อนในการคำนวณ (ครั้ง)					
	บวก	ลบ	คูณ	หาร	ยกกำลัง	รวม
SMLP	2,190	0	2,190	85	85	4,550
QMLP-I	1,923	0	1,921	77	85	4,006
QMLP-II	1,920	0	1,920	76	76	3,992

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางที่ 6.7 พบว่าโครงข่ายประสาทเทียมแบบ QMLP-I และ QMLP-II จะมีความซับซ้อนในการคำนวณขณะแปลความหมาย เมื่อรวมทุกการคำนวณน้อยกว่าโครงข่ายประสาทเทียมแบบ SMLP ประมาณ 11.9% และ 12.3% ตามลำดับ

จากผลการทดลองในตารางที่ 6.5 และ 6.6 จะพบว่าโครงข่ายประสาทเทียมแบบ QMLP-II จะมีค่าความถูกต้องในการรู้จำระดับขนาดหรือปริมาณ (Quantitative Level) ต่ำที่สุด ทั้งในส่วนของการรู้จำระดับขนาดหรือปริมาณ 3 ระดับและ 5 ระดับ ซึ่งสมมุติฐานที่ทำให้วิธีการของโครงข่ายประสาทเทียมแบบ QMLP-II มีค่าความถูกต้องในการรู้จำระดับขนาดหรือปริมาณต่ำกว่าโครงข่ายประสาทเทียมแบบ QMLP-I ก็คือ ขนาดความกว้างของแต่ละช่วงชั้นที่ใช้ในการเปรียบเทียบเพื่อระบุขนาดหรือปริมาณของโครงข่ายประสาทเทียมแบบ QMLP-II มีขนาดความกว้างที่แคบกว่าโครงข่ายประสาทเทียมแบบ QMLP-I เช่น ในกรณีที่ทำมือบอกขนาดหรือปริมาณต้องการรู้จำระดับขนาดหรือปริมาณทั้งหมด 3 ระดับ ขนาดความกว้างของแต่ละช่วงชั้นที่ใช้ในการระบุขนาดหรือปริมาณของโครงข่ายประสาทเทียมแบบ QMLP-II กำหนดได้โดยการแบ่งค่าผลลัพธ์ของ Sigmoid Function เป็น 3 ช่วงชั้นที่เท่า ๆ กัน ซึ่งค่าผลลัพธ์ของ Sigmoid Function มีค่าระหว่าง 0 ถึง 1 ดังนั้นความกว้างของแต่ละช่วงชั้นจะมีค่าประมาณ 0.33 ดังแสดงในรูปที่ 6.5



รูปที่ 6.5 แสดงขนาดความกว้างของแต่ละช่วงชั้นของโครงข่ายประสาทเทียมแบบ QMLP-II

จากรูปที่ 6.5 เป็นรูปที่แสดงการกำหนดขนาดความกว้างของแต่ละช่วงชั้นของโครงข่ายประสาทเทียมแบบ QMLP-II ซึ่งจากโครงสร้างของโครงข่ายประสาทเทียมแบบ QMLP-II ในส่วนของทำมือบอกขนาดหรือปริมาณแต่ละท่า ที่ชั้นผลลัพธ์ (Output Layer) มีโหนดที่เกี่ยวข้องทั้งหมด 2 โหนด แต่ละโหนดก็จะมีวัตถุประสงค์ที่แตกต่างกัน กล่าวคือ โหนดที่หนึ่ง (D_i) จะนำไปใช้ในการเปรียบเทียบกับค่าผลลัพธ์ของทำมืออื่น เพื่อระบุความหมายของทำมือบอกขนาดหรือปริมาณ และโหนดที่สอง (Q_i) จะนำไปใช้ในการระบุขนาดหรือปริมาณของทำมือบอกขนาดหรือปริมาณ จากรูป สามารถอธิบายได้ว่า ไม่ว่าค่าผลลัพธ์ของโหนดที่ใช้ในการระบุความหมาย (D_i) ของทำมือ

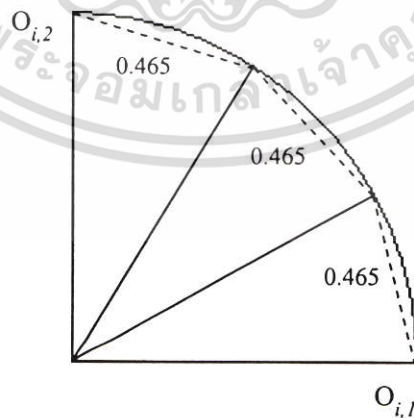
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บอกขนาดหรือปริมาณจะเป็นเท่าไรก็ตาม ความกว้างของแต่ละช่วงชั้นที่แบ่งจากโหนดที่ใช้ในการระบุขนาดหรือปริมาณ (Q_i) ก็ยังมีค่าคงที่

ส่วนกรณีของโครงข่ายประสาทเทียมแบบ QMLP-I ขนาดความกว้างของแต่ละช่วงชั้น จะกำหนดได้โดยการแบ่งค่าองศาทั้งหมด (90 องศา) ออกเป็นช่วงชั้นที่เท่า ๆ กัน ยกตัวอย่างเช่น ในกรณีที่ทำมือบอกขนาดหรือปริมาณต้องการรู้จําระดับขนาดหรือปริมาณทั้งหมด 3 ระดับ ดังนั้นความกว้างของแต่ละช่วงชั้นจะมีค่าเท่ากับ 30 องศา (90/3) เมื่อคำนวณขนาดความกว้างของแต่ละช่วงชั้นโดยใช้สมการที่ 6.14 เพื่อนำไปเปรียบเทียบกับขนาดความกว้างของโครงข่ายประสาทเทียมแบบ QMLP-II จากการคำนวณ ขนาดความกว้างของแต่ละช่วงชั้นของโครงข่ายประสาทเทียมแบบ QMLP-I จะมีค่าประมาณ 0.465 ดังแสดงในรูปที่ 6.6 ซึ่งจะพบว่า โครงข่ายประสาทเทียมแบบ QMLP-I มีขนาดความกว้างของแต่ละช่วงชั้นที่กว้างกว่าแต่ละช่วงชั้นของโครงข่ายประสาทเทียมแบบ QMLP-II ซึ่งเป็นผลทำให้โครงข่ายประสาทเทียมแบบ QMLP-I มีค่าความถูกต้องในการรู้จําขนาดหรือปริมาณที่ดีกว่าโครงข่ายประสาทเทียมแบบ QMLP-II เนื่องจากว่า เมื่อความกว้างของแต่ละช่วงชั้นมีขนาดที่กว้างกว่า ทำให้ความน่าจะเป็นที่ค่าผลลัพธ์ทั้งสองโหนดของทำมือบอกขนาดหรือปริมาณจะข้ามไปอีกช่วงชั้นเป็นไปได้น้อยลง

$$\text{Range} = r \cdot \sin\left(\frac{\theta}{2}\right) \cdot 2 \quad (6.14)$$

กำหนดให้ r คือ รัศมีของวงกลม ในที่นี้มีค่าเท่ากับ 0.9
 θ คือ ขนาดองศาของแต่ละช่วงชั้น



รูปที่ 6.6 แสดงขนาดความกว้างของแต่ละช่วงชั้นของโครงข่ายประสาทเทียมแบบ QMLP-I

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการทดสอบสมมุติฐาน ทำได้โดยการลดค่าองศาที่ใช้ในการระบุขนาดหรือปริมาณของโครงข่ายประสาทเทียมแบบ QMLP-I เป็นผลให้ขนาดความกว้างของแต่ละช่วงชั้นของโครงข่ายประสาทเทียมแบบ แบบ QMLP-I มีขนาดแคบลง ซึ่งผลการทดลอง แสดงในตารางที่ 6.7 และ 6.8

ตารางที่ 6.8 และ 6.9 เป็นผลการทดลองในการตรวจสอบสมมุติฐาน ในกรณี Signer Dependence และ Signer Independence พบว่า ค่าความถูกต้องในการรู้จำขนาดหรือปริมาณของโครงข่ายประสาทเทียมแบบ QMLP-I ทั้ง 2 กรณี มีแนวโน้มที่ลดลง เมื่อลดค่าองศา

ตารางที่ 6.8 แสดงค่าความถูกต้องกรณีที่กำหนดค่าองศาของโครงข่ายประสาทเทียมแบบ QMLP-I เท่ากับ 90 63 และ 55 องศา เปรียบเทียบกับโครงข่ายประสาทเทียมแบบ QMLP-II ในการรู้จำท่ามือบอกขนาดหรือปริมาณในกรณีของ Signer Dependence

Method	Accuracy (%)					
	3 Levels			5 Levels		
	Non-QP Meaning	QP Meaning	QP Level	Non-QP Meaning	QP Meaning	QP Level
QMLP-I 90°	99.28	99.83	99.67	99.28	99.79	96.09
QMLP-I 63°	99.20	99.73	98.86	99.20	99.77	94.72
QMLP-I 55°	99.15	99.80	98.36	99.15	99.67	94.13
QMLP-II	99.20	99.87	98.67	99.20	99.75	94.64

ตารางที่ 6.9 แสดงค่าความถูกต้องกรณีที่กำหนดค่าองศาของโครงข่ายประสาทเทียมแบบ QMLP-I เท่ากับ 90 63 และ 55 องศา เปรียบเทียบกับโครงข่ายประสาทเทียมแบบ QMLP-II ในการรู้จำท่ามือบอกขนาดหรือปริมาณในกรณีของ Signer Independence

Method	Accuracy (%)					
	3 Levels			5 Levels		
	Non-QP Meaning	QP Meaning	QP Level	Non-QP Meaning	QP Meaning	QP Level
QMLP-I 90°	81.70	94.44	97.07	81.70	94.87	92.45
QMLP-I 63°	81.53	94.41	95.18	81.53	94.30	91.18
QMLP-I 55°	81.53	94.22	94.52	81.53	94.24	90.34
QMLP-II	81.67	94.15	94.88	81.67	94.57	90.74

6.2.3 สรุปผลการทดลอง

จากผลการทดลอง พบว่า โครงข่ายประสาทเทียมแบบ SMLP และ QMLP-I มีค่าความถูกต้องในการรู้จำในส่วนของการรู้จำความหมายของท่ามือทั้ง 2 ประเภท ใกล้เคียงกัน ทั้งในกรณีที่ต้องการรู้จำระดับขนาดหรือปริมาณ 3 และ 5 ระดับ ทั้งในการทดลองแบบ Signer Dependence และ Signer Independence แต่ในส่วนของการรู้จำระดับขนาดหรือปริมาณ ในกรณีของ Signer Dependence โครงข่ายประสาทเทียมแบบ QMLP-I จะมีค่าความถูกต้องต่ำกว่าโครงข่ายประสาทเทียมแบบ SMLP ประมาณ 0.25% และ 1.29% ในกรณีที่ต้องการรู้จำระดับขนาดหรือปริมาณ 3 และ 5 ระดับ ตามลำดับ ส่วนกรณีของ Signer Independence จะต่ำกว่าประมาณ 0.72% และ 0.64% ในกรณีที่ต้องการรู้จำระดับขนาดหรือปริมาณ 3 และ 5 ระดับ ตามลำดับ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

สรุปผลการทดลองและข้อเสนอแนะ

จากผลการทดลองในบทที่ 6 ซึ่งสามารถสรุปได้ดังนี้ คือ ในส่วนแรกของงานวิจัย จะเป็นการเปรียบเทียบวิธีการสำหรับรู้จำท่ามือพื้นฐานในภาษามือไทย ในประเด็นของความถูกต้องและความซับซ้อนในการคำนวณขณะแปลความหมาย วิธีการที่เปรียบเทียบคือ โครงข่ายประสาทเทียมแบบ MLP, RBF และ Bayesian Estimator สรุปได้ว่า ในประเด็นของความถูกต้อง วิธีการทั้ง 3 วิธี จะมีค่าความถูกต้องที่ใกล้เคียงกัน คือ 99.25%, 99.23% และ 99.30% ตามลำดับ ในกรณี Signer Dependence และ 83.40%, 83.33% และ 83.75% ตามลำดับ ในกรณี Signer Independence ส่วนประเด็นของความซับซ้อนในการคำนวณขณะแปลความหมาย ที่ระดับความถูกต้องที่ใกล้เคียงกัน โครงข่ายประสาทเทียมแบบ Multilayer Perceptron จะมีความซับซ้อนในการคำนวณขณะแปลความหมายน้อยที่สุด ในส่วนที่สองของงานวิจัย เป็นการเปรียบเทียบการใช้โครงข่ายประสาทเทียมแบบ SMLP กับการใช้โครงข่ายประสาทเทียมแบบ QMLP ในการรู้จำท่ามือพื้นฐานทั้ง 2 แบบ ซึ่งสรุปได้ว่า โครงข่ายประสาทเทียมแบบ QMLP ทั้ง 2 แบบ และ SMLP ในส่วนของการรู้จำความหมายของท่ามือทั้ง 2 ประเภท จะมีค่าความถูกต้องใกล้เคียงกัน ทั้งในการทดลองแบบ Signer Dependence และ Signer Independence และในส่วนของการรู้จำขนาดหรือปริมาณ โครงข่ายประสาทเทียมแบบ SMLP มีค่าความถูกต้องสูงกว่าโครงข่ายประสาทเทียมแบบ QMLP-I ในกรณีที่ต้องการรู้จำขนาดหรือปริมาณ 3 และ 5 ระดับ ประมาณ 0.25% และ 1.29% ในการทดลองแบบ Signer Dependence และสูงกว่าประมาณ 0.72% และ 0.64% ในการทดลองแบบ Signer Independence และสูงกว่าโครงข่ายประสาทเทียมแบบ QMLP-II ในกรณีที่ต้องการรู้จำขนาดหรือปริมาณ 3 และ 5 ระดับ ประมาณ 1.25% และ 2.74% ในการทดลองแบบ Signer Dependence และสูงกว่าประมาณ 2.90% และ 2.35% ในการทดลองแบบ Signer Independence แต่โครงข่ายประสาทเทียมแบบ QMLP จะมีข้อดี ซึ่งสรุปได้ดังนี้

- ปริมาณข้อมูลของท่ามือบอกขนาดหรือปริมาณที่ใช้สำหรับการเรียนรู้ต่ำกว่า
- ในขั้นตอนของการรู้จำสามารถรู้จำระดับขนาดหรือปริมาณได้มากกว่าระดับขนาดหรือปริมาณในขั้นตอนของการเรียนรู้
- ความซับซ้อนในการคำนวณขณะแปลความหมายน้อยกว่า

งานที่จะพัฒนาต่อไป คือ การนำโครงข่ายประสาทเทียมแบบ QMLP-I ไปทดลองใช้กับท่ามือบอกขนาดหรือปริมาณที่เป็นท่าที่ต้องใช้มือทั้งสองข้างในการบอกขนาดหรือปริมาณ และท่ามือบอกขนาดหรือปริมาณที่มีการใช้ตำแหน่งของมือในการระบุขนาดหรือปริมาณ

เอกสารอ้างอิง

- [1] C. Shahabi, L. Kaghazian, S. Mehta, A. Choting, G. Shanbhag and M. McLaughlin. "Analysis of Haptic Data for Sign Language Recognition" **9th International Conference on Human Computer Interaction, New Orleans, Aug. 2001.**
- [2] Worawit Werapan, Nopporn Chotikakamthorn. "Gesture Segmentation Technique for Thai Sign Language Translation" **2nd International Symposium on Communications and Information Technology, Chonburi, Thailand, October, 2002.** pp. 483-486.
- [3] Manjula B. Waldron, Soowon Kim. "Isolated ASL Sign Recognition System for Deaf Persons" **IEEE Transactions on Rehabilitation Engineering**, vol. 3, no. 3, Sep. 1995.
- [4] S. Sidney Fels, Geoffrey E. Hinton. "Glove-Talk: A Neural Network Interface Between a Data-Glove and a Speech Synthesizer" **IEEE Transactions on Neural Network**, vol. 4, no. 1, January 1993.
- [5] P. Vamplew. "Recognition of Sign Language Gestures using Neural Network" **1st Euro. Conference on Virtual Reality and Associate Technology, Maidenhead, UK, 1996.** pp. 27-33.
- [6] Morton Nadler, Eric P. Smith. **Pattern Recognition Engineering.** John Wiley & Sons, Inc. 1993.
- [7] W. Philpot. **Digital Image Processing.** Cornell University, Jan. 2001.
- [8] James A. Freeman, David M. Skapura. **Neural Network Algorithm Application and Programming Techniques.** Addition-Wesley publishing Company, Inc. 1991.
- [9] Martin T. Hagan, Howard B. Demuth, Mark Beale. **Neural Network Design.** An International Thompson Publishing Company, Inc. 1996.
- [10] D.E. Rumelhart et al. "Parallel Distributed Processing : Explorations in the Micro Structure of Cognition Vol. I : Foundations" **MIT Press, Cambridge, MA, 1986.**
- [11] M. J. L. Orr. "Introduction to Radial Basis Function Networks" **Technical Report April, Center for Cognitive Science, Univ. of Edinburgh, April 1996.**
- [12] สมาคมคนหูหนวกแห่งประเทศไทย. **ปทานุกรมภาษามือไทย ฉบับปรับปรุงและขยายเพิ่มเติม.** ไทยวัฒนาพานิช, 1990.
- [13] Virtual Technologies, Inc. **VirtualHand[®] Software Library Reference Manual.** Aug. 5, 1998.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- [14] Kessler G., Hodges L., Walker N. "Evaluation of the CyberGlove™ as a whole Hand Input Device" **ACM Transactions on Computer-Human Interaction**,
- [15] LaViola J., "A Survey of Hand Posture and Gesture Recognition Techniques and Technology" **Technical Report CS-99-11, Brown University, Department of Computer Science, Providence RI, June 1999.**



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

สมการที่ใช้สำหรับคำนวณความซับซ้อนขณะแปลความหมาย

ในภาคผนวก ก. นี้ เป็นการอธิบายถึงสมการที่ใช้สำหรับการแปลความหมายของข้อมูล ของวิธีการที่ได้ทำการเปรียบเทียบ ซึ่งก็คือ โครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron (MLP) โครงข่ายประสาทเทียมแบบ Radial Basis Function (RBF) และวิธีการแบบ Bayesian Estimator รวมทั้งความซับซ้อนในการคำนวณขณะแปลความหมายของแต่ละวิธี

ก.1 โครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron (MLP)

กำหนดให้

x คือ เวกเตอร์ข้อมูลที่ต้องการรู้จำ ขนาด n มิติ

W_{ji} คือ ค่า Weight ที่เชื่อมจากโหนด i^{th} ของชั้นอินพุตไปยังโหนด j^{th} ของชั้นซ่อน โดยที่ $i = 1, 2, \dots, n$ และ $j = 1, 2, \dots, s$ ดังนั้น W_{ji} มีขนาด $s \times n$

Z_{kj} คือ ค่า Weight ที่เชื่อมจากโหนด j^{th} ของชั้นซ่อนไปยังโหนด k^{th} ของชั้นผลลัพธ์ โดยที่ $j = 1, 2, \dots, s$ และ $k = 1, 2, \dots, m$ ดังนั้น Z_{kj} มีขนาด $m \times s$

เมื่อ

n คือ จำนวนโหนดในชั้นอินพุต

s คือ จำนวนโหนดในชั้นซ่อน

m คือ จำนวนโหนดในชั้นผลลัพธ์

ขั้นตอนการแปลความหมายของโครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron (MLP) สามารถสรุปได้ดังนี้

1. เมื่อเวกเตอร์ข้อมูล (x) ถูกส่งเข้าสู่โครงข่ายโดยผ่านชั้นอินพุต (Input Layer) และจะถูกส่งต่อไปยังโหนดแต่ละโหนดในชั้นซ่อน โดยผ่านค่า Weight (W_{ji}) เพื่อคำนวณหาค่าผลลัพธ์ของแต่ละโหนด โดยใช้สมการที่ ก.1 และ ก.2

$$\text{net}_j = \sum_{i=1}^n W_{ji}x_i \quad (\text{ก.1})$$

$$h_j = \frac{1}{1 + \exp(-\text{net}_j)} \quad (\text{ก.2})$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมการที่ ข.1 เป็นสมการของการคำนวณหาค่า net input ของโหนด j^{th} ในชั้นซ่อน (Hidden Layer) เพื่อส่งผ่านไปยังสมการที่ ก.2 ซึ่งเป็นสมการของ Sigmoid Function เพื่อคำนวณหาค่าผลลัพธ์ของโหนด j^{th} ในชั้นซ่อน ซึ่งสมการทั้ง 2 สมการนี้ สามารถคำนวณความซับซ้อนในการแปลความหมายของโหนด j^{th} ในชั้นซ่อน สรุปได้ดังนี้

$$\text{การบวก} = (n-1)+1 \quad (\text{ก.3})$$

$$\text{การคูณ} = n \quad (\text{ก.4})$$

$$\text{การหาร} = 1 \quad (\text{ก.5})$$

$$\text{ยกกำลัง} = 1 \quad (\text{ก.6})$$

ถ้าต้องการหาความซับซ้อนในการคำนวณของทุกโหนดในชั้นซ่อน ก็ทำได้โดยการนำจำนวนโหนดทั้งหมดในชั้นซ่อน (s) ไปคูณกับสมการข้างต้น ก็จะได้การคำนวณของทุกโหนดในชั้นซ่อน

2. เมื่อได้ผลลัพธ์ที่ชั้นซ่อนแล้ว ผลลัพธ์ที่ชั้นซ่อนก็จะถูกส่งต่อไปยังโหนดแต่ละโหนดในชั้นผลลัพธ์ โดยผ่านค่า Weight (Z_{kj}) เพื่อคำนวณหาค่าผลลัพธ์ของแต่ละโหนด โดยใช้สมการที่ ก.7 และ ก.8

$$\text{net}_k = \sum_{j=1}^s Z_{kj} h_j \quad (\text{ก.7})$$

$$o_k = \frac{1}{1 + \exp(-\text{net}_k)} \quad (\text{ก.8})$$

สมการที่ ก.7 เป็นสมการของการคำนวณหาค่า net input ของโหนด k^{th} ในชั้นผลลัพธ์ (Output Layer) เพื่อส่งผ่านไปยังสมการที่ ก.8 ซึ่งเป็นสมการของ Sigmoid Function เพื่อคำนวณหาค่าผลลัพธ์ของโหนด k^{th} ในชั้นผลลัพธ์ ซึ่งสมการทั้ง 2 สมการนี้ สามารถคำนวณความซับซ้อนในการแปลความหมายของโหนด k^{th} ในชั้นผลลัพธ์ สรุปได้ดังนี้

$$\text{การบวก} = (s-1)+1 \quad (\text{ก.9})$$

$$\text{การคูณ} = s \quad (\text{ก.10})$$

$$\text{การหาร} = 1 \quad (\text{ก.11})$$

$$\text{ยกกำลัง} = 1 \quad (\text{ข.12})$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าต้องการหาความซับซ้อนในการคำนวณของทุกโหนดในชั้นผลลัพธ์ ก็ทำได้โดยการนำจำนวนโหนดทั้งหมดในชั้นผลลัพธ์ (m) ไปคูณกับสมการข้างต้น ก็จะได้การคำนวณของทุกโหนดในชั้นผลลัพธ์

ก.2 โครงข่ายประสาทเทียมแบบ Radial Basis Function (RBF)

กำหนดให้

- \mathbf{x} คือ เวกเตอร์ข้อมูลที่ต้องการรู้จำ ขนาด n มิติ
- \mathbf{Z}_{kj} คือ ค่า Weight ที่เชื่อมจากโหนด j^{th} ของชั้นซ่อนไปยังโหนด k^{th} ของชั้นผลลัพธ์ โดยที่ $j = 1, 2, \dots, s$ และ $k = 1, 2, \dots, m$ ดังนั้น \mathbf{Z}_{kj} มีขนาด $m \times s$
- μ_j คือ ค่า Center ของโหนด j^{th} ในชั้นซ่อน ซึ่งจะมีขนาดเท่ากับขนาด n มิติ
- σ_j^2 คือ ค่าความแปรปรวน หรือค่า Width ของ Center โหนด j^{th} ในชั้นซ่อน

เมื่อ

- n คือ จำนวนโหนดในชั้นอินพุต
- s คือ จำนวนโหนดในชั้นซ่อน
- m คือ จำนวนโหนดในชั้นผลลัพธ์

ขั้นตอนการแปลความหมายของโครงข่ายประสาทเทียมแบบ Radial Basis Function (RBF) สามารถสรุปได้ดังนี้

1. เมื่อเวกเตอร์ข้อมูล (\mathbf{x}) ถูกส่งเข้าสู่โครงข่ายโดยผ่านชั้นอินพุต (Input Layer) และถูกส่งต่อไปยังแต่ละโหนดในชั้นซ่อน โดยผ่านค่า Center (μ_j) เพื่อคำนวณหาค่าผลลัพธ์ของแต่ละโหนด โดยใช้สมการที่ ก.13

$$h_j = \exp\left(-\frac{\|\mathbf{x} - \mu_j\|^2}{\sigma_j^2}\right) \quad (\text{ก.13})$$

สมการที่ ก.13 เป็นสมการของ Radial Basis Function โดยที่ค่า net input ของฟังก์ชันนี้คือ ค่า Euclidean Distance ระหว่างเวกเตอร์ข้อมูล (\mathbf{x}) กับค่า Center (μ_j) ของโหนด j^{th} ในชั้นซ่อน (Hidden Layer) ซึ่งสมการนี้ สามารถคำนวณความซับซ้อนในการแปลความหมายของโหนด j^{th} ในชั้นซ่อน สรุปได้ดังนี้

$$\text{การบวก} = (n-1) \quad (\text{ก.14})$$

$$\text{การลบ} = n \quad (\text{ก.15})$$

$$\text{การหาร} = 1 \quad (\text{ก.16})$$

$$\text{ยกกำลัง} = (n+1) \quad (\text{ก.17})$$

ถ้าต้องการหาความซับซ้อนในการคำนวณของทุกโหนดในชั้นซ่อน ก็ทำได้โดยการนำจำนวนโหนดทั้งหมดในชั้นซ่อน (s) ไปคูณกับสมการข้างต้น ก็จะได้การคำนวณของทุกโหนดในชั้นซ่อน

2. เมื่อได้ผลลัพธ์ที่ชั้นซ่อนแล้ว ผลลัพธ์ที่ชั้นซ่อนก็จะถูกส่งต่อไปยังโหนดแต่ละโหนดในชั้นผลลัพธ์ โดยผ่านค่า Weight (Z_{kj}) เพื่อคำนวณหาค่าผลลัพธ์ของแต่ละโหนด โดยใช้สมการที่ ก.18

$$o_k = \sum_{j=1}^s Z_{kj} h_j \quad (\text{ก.18})$$

สมการที่ ก.18 เป็นสมการของการคำนวณหาค่า net input ของโหนด k^{th} ในชั้นผลลัพธ์ (Output Layer) เนื่องจากว่าที่ชั้นนี้จะใช้ Linear Function ซึ่งค่าผลลัพธ์ของฟังก์ชันจะมีค่าเท่ากับค่า net input ดังนั้นสมการที่ ก.18 จึงกลายเป็นค่าผลลัพธ์ของโหนด k^{th} ในชั้นผลลัพธ์ ซึ่งสามารถคำนวณความซับซ้อนในการแปลความหมายของโหนด k^{th} ในชั้นผลลัพธ์ สรุปได้ดังนี้

$$\text{การบวก} = (s-1) \quad (\text{ข.19})$$

$$\text{การคูณ} = s \quad (\text{ข.20})$$

ถ้าต้องการหาความซับซ้อนในการคำนวณของทุกโหนดในชั้นผลลัพธ์ ก็ทำได้โดยการนำจำนวนโหนดทั้งหมดในชั้นผลลัพธ์ (m) ไปคูณกับสมการข้างต้น ก็จะได้การคำนวณของทุกโหนดในชั้นผลลัพธ์

ก.3 Bayesian Estimator

สมการที่ใช้สำหรับคำนวณความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการแบบ

Bayesian Estimator จะใช้สมการที่ ก.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$d_i(\mathbf{x}) = (\mathbf{x} - \boldsymbol{\mu}_i)' \mathbf{C}_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i) + \ln|C_i| \quad (\text{ก.21})$$

โดยที่

- \mathbf{x} คือ เวกเตอร์ข้อมูล ขนาด n มิติ
- $\boldsymbol{\mu}_i$ คือ ค่าเฉลี่ยของคลาส i^{th} ขนาด n มิติ
- \mathbf{C}_i^{-1} คือ เมทริกซ์ส่วนกลับของค่าความแปรปรวนร่วม (Inverse Covariance) ของคลาส i^{th} ขนาด $n \times n$ มิติ
- C_i คือ ค่า determinant ของค่าความแปรปรวนร่วม คลาส i^{th}

จากสมการที่ ก.21 เป็นสมการของการกระจายแบบปกติ (Normal Distribution) ของข้อมูล ซึ่งขั้นตอนของการแปลความหมายของวิธีการแบบ Bayesian Estimator คือ ในเทอมแรกของสมการที่ ก.21 คือ $(\mathbf{x} - \boldsymbol{\mu}_i)' \mathbf{C}_i^{-1} (\mathbf{x} - \boldsymbol{\mu}_i)$ เป็นเทอมที่คำนวณหาระยะทางระหว่างเวกเตอร์ข้อมูล (\mathbf{x}) กับค่าเฉลี่ย ($\boldsymbol{\mu}_i$) ของคลาสที่ i^{th} แล้วนำไปคูณกับค่าส่วนกลับของค่าความแปรปรวนร่วมของคลาสที่ i^{th} (\mathbf{C}_i^{-1}) แล้วนำผลลัพธ์ที่ได้ไปบวกกับเทอมที่สองของสมการ คือค่า ลอการิทึมของค่า Determinant ของค่าความแปรปรวนร่วมของคลาสที่ i^{th} ซึ่งความซับซ้อนในการคำนวณสรุปได้ดังนี้ คือ

$$\text{การบวก} = (n-1)n + n \quad (\text{ก.22})$$

$$\text{การลบ} = n \quad (\text{ก.23})$$

$$\text{การคูณ} = (n^2 + n) \quad (\text{ก.24})$$

ความซับซ้อนในสมการที่ ก.22 - ก.24 เป็นความซับซ้อนในการคำนวณขณะแปลความหมายของคลาสที่ i^{th} เพียงคลาสเดียวเท่านั้น ซึ่งจะต้องมีการคำนวณหาความซับซ้อนของทุกคลาสที่ต้องการเปรียบเทียบ ทำได้โดยการนำจำนวนคลาสทั้งหมด (m) ไปคูณกับสมการข้างต้น ก็จะได้การคำนวณของทุกคลาสที่ต้องการเปรียบเทียบ

สมการที่ ก.21 เป็นสมการของการกระจายแบบปกติที่ได้นำไปใช้ในงานวิจัยนี้ จะพบว่าในเทอมที่ 1 ซึ่งเป็นเทอมหลักของสมการมีการคำนวณมากที่สุด ในทางปฏิบัติทั่วไป พยายามที่จะลดความซับซ้อนของสมการ เพื่อให้การทำงานมีความรวดเร็วขึ้น ซึ่งวิธีการที่สามารถลดความซับซ้อนของสมการได้ คือ การใช้ทฤษฎีของ Cholesky factorization กล่าวคือ เนื่องจากว่าค่าส่วนกลับของค่าความแปรปรวนร่วม (\mathbf{C}_i^{-1}) ของทุกคลาสจะมีคุณลักษณะที่สมมาตร (Symmetry) กัน ซึ่งทฤษฎีของ Cholesky factorization จะแยกค่าส่วนกลับของค่าความแปรปรวนร่วม (\mathbf{C}_i^{-1}) ออกเป็น 2 เมทริก คือ

$$C_i^{-1} = (C_i^{-1/2})' C_i^{-1/2} \quad (ก.25)$$

โดยที่

$C_i^{-1/2}$ คือ เมตริกสามเหลี่ยมบน (upper triangular matrix)

ดังนั้นสมการที่ ก.21 สามารถเขียนใหม่ได้ คือ

$$d_i(x) = (x - \mu_i)' (C_i^{-1/2})' C_i^{-1/2} (x - \mu_i) + \ln|C_i| \quad (ก.26)$$

จากสมการที่ ก.26 ค่าผลลัพธ์ของ $(x - \mu_i)' (C_i^{-1/2})'$ กับ $C_i^{-1/2} (x - \mu_i)$ มีค่าเท่ากัน กล่าวคือ

$$(x - \mu_i)' (C_i^{-1/2})' = (C_i^{-1/2} (x - \mu_i))' \quad (ก.27)$$

ดังนั้น ในเทอม $(x - \mu_i)' (C_i^{-1/2})' C_i^{-1/2} (x - \mu_i)$ ของสมการที่ ก.26 จะมีการคำนวณในส่วนของ $C_i^{-1/2} (x - \mu_i)$ เท่านั้น ดังสมการที่ ก.28

$$v_i = C_i^{-1/2} (x - \mu_i) \quad (ก.28)$$

ซึ่งสมการที่ ก.26 สามารถเขียนใหม่ได้ดังสมการที่ ก.29

$$d_i(x) = (v_i)' v_i + \ln|C_i| \quad (ก.29)$$

ผลลัพธ์ของสมการที่ ก.29 เป็นผลลัพธ์ของคลาสที่ i^{th} ซึ่งสามารถสรุปความซับซ้อนในการคำนวณได้ดังนี้

$$\text{การบวก} = \frac{(n-1)n}{2} + n \quad (ก.30)$$

$$\text{การลบ} = n \quad (ก.31)$$

$$\text{การคูณ} = \frac{n(n+1)}{2} + n \quad (ก.32)$$

ความซับซ้อนในสมการที่ ก.30 - ก.32 เป็นความซับซ้อนในการคำนวณขณะแปลความหมายของคลาสที่ i^{th} เพียงคลาสเดียวเท่านั้น ถ้าต้องการคำนวณความซับซ้อนของทุกคลาส ทำได้โดยการเอกสทรานี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำจำนวนคลาสทั้งหมด (m) ไปคูณกับสมการข้างต้น ก็จะได้การคำนวณของทุกคลาสที่ต้องการเปรียบเทียบ

จากสมการการคำนวณความซับซ้อนของทั้งโครงข่ายประสาทเทียมและวิธีการแบบ Bayesian Estimator สามารถสรุปเป็นสมการ ได้ดังตาราง ก.1

ตารางที่ ก.1 สมการที่ใช้ในการคำนวณความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการที่เปรียบเทียบ

วิธีการ	ความซับซ้อนในการคำนวณ (ครั้ง)				
	บวก	ลบ	คูณ	หาร	ยกกำลัง
MLP	$(n-1)s + s + (s-1)m + m$	0	$(n+m)s$	$s+m$	$s+m$
RBF	$(n-1)s + (s-1)m$	sn	ms	s	$(n+1)s$
BAYE	$[(n-1)n + n]m$	nm	$(n^2 + n)m$	0	0
BAYE (Cholesky)	$\left(\frac{(n-1)n}{2} + n\right)m$	nm	$\left(\frac{n(n+1)}{2} + n\right)m$	0	0

ถ้าลองคำนวณความซับซ้อนของวิธีการที่เปรียบเทียบกัน โดยใช้สมการในตารางที่ ก.1 ผลที่ได้แสดงในตารางที่ ก.2

ตารางที่ ก.2 แสดงความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการที่ใช้ในการเปรียบเทียบในกรณี Signer Dependence

วิธีการ	ความซับซ้อนในการคำนวณ (ครั้ง)					
	บวก	ลบ	คูณ	หาร	ยกกำลัง	รวม
MLP	1,160	0	1,160	60	60	2,440
RBF	2,240	720	1,600	40	760	5,360
BAYE	12,960	720	13,680	0	0	27,360
BAYE (Cholesky)	6,840	720	7,560	0	0	15,120

ตารางที่ ก.2 แสดงผลการคำนวณความซับซ้อนในการแปลความหมายของวิธีการที่เปรียบเทียบ โดยใช้วิธีการทดลองจากบทที่ 6 ในกรณีของ Signer Dependence (ตารางที่ 6.3)

ตารางที่ ก.3 แสดงความซับซ้อนในการคำนวณขณะแปลความหมายของวิธีการที่ใช้ในการ
เปรียบเทียบในกรณี Signer Independence

วิธีการ	ความซับซ้อนในการคำนวณ (ครั้ง)					
	บวก	ลบ	คูณ	หาร	ยกกำลัง	รวม
MLP	1,740	0	1,740	70	70	3,620
RBF	2,525	810	1,800	45	855	6,035
BAYE	12,960	720	13,680	0	0	27,360
BAYE (Cholesky)	6,840	720	7,560	0	0	15,120

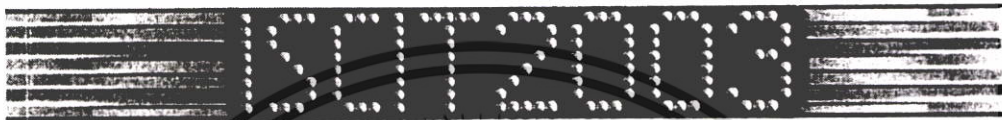
ตารางที่ ก.3 แสดงผลการคำนวณความซับซ้อนในการแปลความหมายของวิธีการที่เปรียบเทียบ
โดยใช้วิธีการทดลองจากบทที่ 6 ในกรณีของ Signer Independence (ตารางที่ 6.4)

จากตารางที่ ก.2 และ ก.3 พบว่าวิธีการแบบ Bayesian Estimator ถ้าใช้ทฤษฎี Cholesky
factorization จะทำให้วิธีการแบบ Bayesian Estimator มีการคำนวณน้อยลงประมาณ 44.7% เมื่อ
เปรียบเทียบกับ Bayesian Estimator ที่ไม่ใช่ทฤษฎี Cholesky factorization

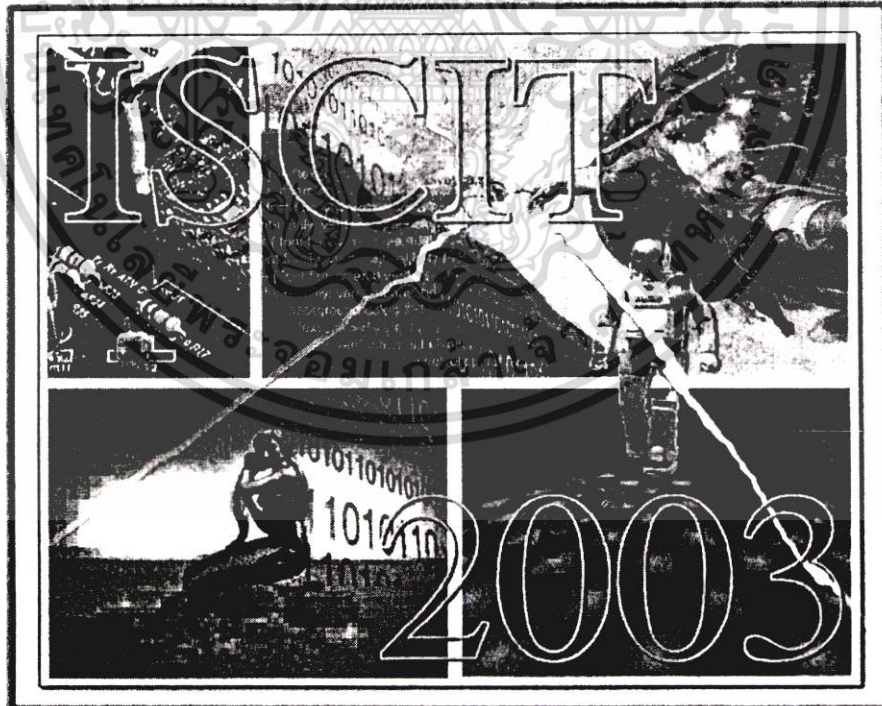
ภาคผนวก ข.

บทความและผลงานวิจัยที่ได้รับการตีพิมพ์

1. Kumphol Wasanapongpan and Nopporn Chotikakamthorn. "A Fast Retrain Neural Network Using Parallel Multi-layer Perceptron for Thai Sign Language Recognition" 3rd ISCIT Conference 2003, pp. 725 – 728



Volume II
Proceedings
The Third International Symposium
on Communications and Information Technologies
September 3-5, 2003
BP Samila Beach Hotel and Resort, Songkhla, Thailand



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



3-5 September 2003
BP Samila Beach Hotel and Resort, Songkhla, Thailand

ISCIT 2003

www.coe.psu.ac.th/iscit2003
The 3rd International Symposium
On Communications and Information Technologies

Official Receipt

Receipt No : ISCIT 2003/114

Date : 29 August 2003

Receive from : Mr. Wasanapongpan Kumphol
Faculty of Information Technology and
The Research Center for King
Mongkut's Institute of Technology
Ladkrabang Technology, Thailand

Paper reference number : 0000275

Your registration details are shown below:

Registration Items	Amount (THB)
Conference Registration:	6,000
Optional Charge: Extra Set(s) of Proceedings:	
Total Amount Due	6,000

Thank you for registration/participating in the ISCIT2003, September 3-5, 2003
BP Samila Beach Hotel and Resort, Songkhla, Thailand

S. Komolthong

 Assoc. Prof. Sinchai Kamolthong
 ISCIT2003 General Secretary

ADVISORY BOARD
 Pichit Kongsangwanna
 Wullon Saitpracha
 Supachok Wittayapoch
 Sivasak Tammarana

STEERING COMMITTEE
 Wanlop Sulamiron
 (Chair), Thailand
 Yoshikazu Miyasaga, Japan
 Hoyai Ngoc Phien, Thailand
 Jae Seon Park, Korea
 Kazi Ahmed, Thailand
 Rokuya Ishii, Japan
 Yong-Hwan Lee, Korea
 Pansak Siriruchanpong, Thailand
 Yoshitoh, Japan
 Chiochuan Luan, Thailand
 Sarwan, India
 David Neo Chek Ling, Malaysia
 Toshio Watanabe, Japan
 Gyan Prasad, India
 Zhou Zheng, China
 Somsak Chiochuan, (SC) Thailand

ORGANIZING COMMITTEE
 General Chair
 Pansak Siriruchanpong, NECTEC
 Technical Program Chairs
 Sung Hui Hong, NUS
 Kiat Charnongchai, TUED
 Tutorial/Special Chairs
 Chusak Limkikul, WU
 Sureswaran Kupadass, USM
 Wengsak Kurutach, MUT
 Local Arrangement Chairs
 Pichaya Indayya, PSU
 Pichanok Surysachai, PSU
 Publication Chairs
 Sittirong Witsurong, PSU
 Tachchai Engchan, PSU
 Publicity Chairs
 Weernaporn Maitasarn, PSU
 Nuttaka Homay, KU
 Kij Rithana, (C) Mahamud, ULA
 Somsak Maitasarn, PSU
 Exhibition Chairs
 Montri Kerdjansong, PSU
 Moon Eun Kyu, CUH
 Thongchai Kamolthong
 Kinwadee Chair
 Thanate Khonadapong, PSU
 Sateen Triumthalerd, NECTEC
 General Secretary
 Sinchai Kamolthong, PSU



ISCIT 2003 Secretariat, Department of Computer Engineering, Faculty of Engineering, Prince of Songkla University, Hatyai, Songkhla, Thailand 90112
 Tel: +66 74 28 7352, Fax: +66 74 212 895 E-mail: iscit2003@coe.psu.ac.th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A Fast Retrain Neural Network Using a Parallel Multi-layer Perceptron for Thai Sign Language Recognition

Kumphol Wasanapongpan and Nopporn Chotikakamthorn

Faculty of Information Technology &
The Research Center for Communications and Information Technology (ReCCIT)
King Mongkut's Institute of Technology Ladkrabang
Chalongkrung Rd., Ladkrabang Bangkok, THAILAND 10520
Email : s3067019@kmitl.ac.th

Abstract

In this paper, a problem of Thai sign language recognition using a neural network is considered. A static posture of Thai sign is captured by using an electronic glove device. The data obtained is fed into a multi-layer Perceptron neural network. The paper addresses the problem of this type of neural net in dealing with re-training, when new posture is added to the system. A special form of a multi-layer Perceptron with two parallel neural networks is proposed to reduce the need for retraining the whole network. The relationship between this neural network architecture and the standard Bayesian estimator is identified and explained. Experimental results with real data are included with some discussion.

1. Introduction

Neural networks have been applied to gesture recognition problems, such as [1] where a neural net was applied for isolated American sign language (ASL) recognition. Their system obtained gesture data from an electronic (data) glove and achieved the accuracy level of 86%. In addition, [2] developed a gesture-recognition system called Glove-Talk, using a neural network for linking information from a data glove and a speech synthesizer. In [3], a neural network was applied to recognition hand postures of English alphabets. With a total of 5 different hand postures, they achieved the accuracy of 96%. In [4], a problem of segmenting continuous Thai signs is considered. The Bayesian estimator is applied in the paper for sign recognition.

One of the most popular neural network architecture is a multi-layer Perceptron. However, when a new pattern needs to be included in the system, the network needs to retrain not just for the new pattern, but all patterns already existed. More importantly, the whole neural weights involve in retraining. This is a serious drawback, specially for large-pattern recognition systems. In this paper, we describe a neural network with two parallel Perceptrons.

The architecture is inspired by the relationship between a Perceptron and a Bayesian estimator. The proposed neural network retains efficient computation as compared with a standard Bayesian estimator, while allowing for partial training of the network.

The paper is organized as follows. First, a theory and notation of Bayesian estimator and neural network are introduced in Section 2. Next, the proposed neural network with its relationship to the Bayesian estimator is described. Experiment results, obtained from a real measurement, are provided with some discussion in Section 3. A conclusion remark is given in Section 4.

2. Standard pattern recognition methods

Bayesian estimator

One of standard and versatile statistical methods for pattern recognition problems is the Bayesian estimator. The Bayesian estimator with a Normal distribution assumption is described by the following equation.

$$d_i(x) = -\frac{1}{\sqrt{2\pi}|C_i|} \exp\left[-\frac{1}{2}(x-\mu_i)^T C_i^{-1}(x-\mu_i)\right] \quad (1)$$

where

- $d_i(x)$ = the discrimination function for class i
- X = the input pattern vector
- μ_i = the mean of the i^{th} class pattern.
- C_i^{-1} = the determinant of the covariance matrix corresponding to the pattern of class i .
- C_i = the inverse covariance of pattern class i .

Note that here a Normal distribution assumption is applied to simplify the discussion. A more general distribution model can be handled by using, for example, a Gaussian mixture model.

The above equation can be simplified by using the logarithmic version, which is described by the following equation.

$$d_i(x) = -\frac{1}{2} \ln(2\pi) - \frac{1}{2} \ln|C_i| - \frac{1}{2} (x - \mu_i)' C_i^{-1} (x - \mu_i) \quad (2)$$

Note that the first term in the above equation does not depend on any class parameter, and can be dropped in practice. Thus, a simplified equivalent estimator is given by

$$d_i(x) = \ln|C_i| + (x - \mu_i)' C_i^{-1} (x - \mu_i) \quad (3)$$

Multi-layer Perceptron neural network

A general structure of a multi-layer Perceptron neural network is shown in Figure 1.

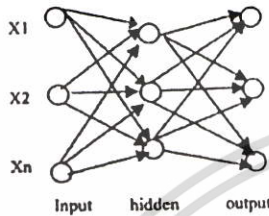


Figure 1. Standard multilayer Perceptron neural network

A neural network of Figure 1 has the following structures

- It is composed of multiple layers, namely an input layer, one or more hidden layer, and the output layer respectively.
- It is fully connected, implying that all nodes in the foregoing layer are connected to every node in the subsequent layer, through a network weight.

From Figure 1, given that X is the vector of n dimension. It is fed to the network input layer. The i^{th} input layer is connected to the j^{th} node of the first hidden layer through the weights W_{ij} , $i = 1, 2, \dots, n$, and $j = 1, 2, \dots, h$. The output of the first hidden layer is obtained by passing the summation of the multiplication between the inputs and the first layer weights, through the activation function, as described by

$$Y = f(x * W) \quad (4)$$

where

- $f()$ = Activate function or transfer function
- Y = layer's output

Next, the output from the first hidden layer is fed as an input to the next layer.

Perhaps the most popular algorithm for training such a network is a Backpropagation algorithm. The algorithm is composed of the forward phase, and the backward phase. It is an efficient way of implementing a Widrow-Hopf LMS learning rule.

3. A parallel multi-layer Perceptron neural network

In this section, a parallel multi-layer Perceptron is derived. The architecture is motivated by the need to incrementally perform a network training for new posture pattern. It is based on the decomposition of a Bayesian estimator. The decomposed parts are computed by using a neural network. To derive the parallel network, notice from Eq. (3) that the first term is independent of the input data. This is equivalent to the offset input of the neural network. In addition, the second term in the same equation, which is $(x' - \mu_i)' C_i^{-1} (x - \mu_i)$, known also the Mahalanobis Distance, can be rewritten as follows.

$$x' C_i^{-1} x - 2\mu_i' C_i^{-1} x + \mu_i' \mu_i \quad (5)$$

where $i = 1, \dots, N_c$, N_c = number of all classes.

From the above equation, it is seen that only the first term is non-linear while the subsequent terms contain only a linear part. It is known that a single-layer neural network is sufficient to model a linear equation. Therefore, based on this finding, a neural network is developed such that it is composed of two parallel networks. The first network is a multi-layer Perceptron with single hidden layer, to model the first non-linear term of Eq. (5). The second network is a single-layer Perceptron, to model the remaining linear terms of Eq. (5). The structure of this parallel network is shown in Figure 2.

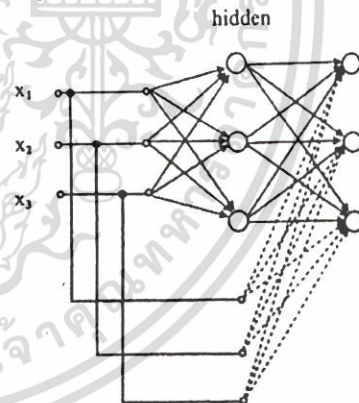


Figure 2. The parallel multi-layer Perceptron network

By inspection, it was found that with sufficient number of known patterns, the inverse covariance matrix of a new pattern can be sufficiently approximated by the combination of the matrices of the existing patterns. Thus, for each new pattern to be included in the system, the network of Figure 2 is incrementally expanded in the manner as shown in Figure 3.

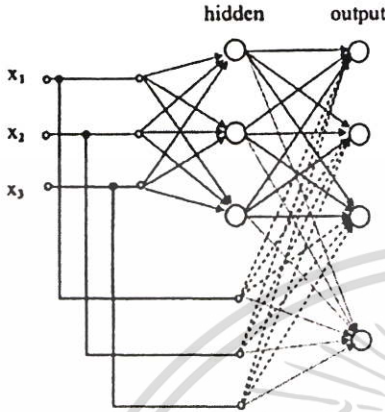


Figure 3. The parallel multi-layer Perceptron when a new pattern is added to the system

To retrain the network of Figure 3, only the weights associated with the new output nodes (shown with dot lines in the figure) need to be calculated. Thus, the retraining process can be speed up. Note, however, in our experiment, it was found that weights of the secondary network (strong dashed lines in Figure 3), connecting inputs to the network outputs, did not have noticeable impact on the overall performance. Those weights are thus may be omitted in practice.

4. Experiment

In this section, results from the experiment with real data are described. In the experiment, data was collected from 5 signers, each performing a total of 35 different static postures in Thai sign language, giving a total of 460 data sets for each pattern (see Figure 4 for example). Each pattern is composed of 22 data values obtained from an electronic glove.

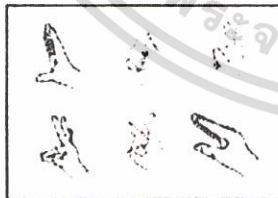


Figure 4. Example of postures in Thai sign language

No. of hidden nodes	10	20	40	70	80
Accuracy (%)	98.65	98.89	99.58	99.67	99.75

Table 1. Recognition accuracy with different number of hidden nodes

Data as collected was used to train a standard multiplayer Perceptron neural network with different number of hidden nodes. The result is shown in Table 1. From the table, it is found that the accuracy is comparable in all cases, when the number of nodes is greater than 20.

Next, another experiment was performed with the Parallel MLP. Thirty-five sign postures were divided into two groups. The first group was used to train the first (primary network) of the two parallel networks. Then, the second group was applied for training the other (secondary) network. Two different training conditions were applied to the network.

1. Parallel MLP with 'equal threshold': Under this training condition, the same desired network outputs are the same for training the primary and secondary networks. For example, the desired response of the output corresponding to a particular input pattern was 0.9, while the desired response for other outputs was 0.1.
2. Parallel MLP with 'compensated threshold': Under this training condition, for the primary network training, the desired response of the output corresponding to a particular input pattern was 0.8, while the desired response for other outputs was 0.2. For the secondary network training, the desired response of the output corresponding to a particular new input pattern was 0.9, while the desired response for other outputs was 0.1. For existing patterns, the desired response for all secondary network outputs was set to 0.

Result from the experiments with different numbers of added sign postures is shown in Table 2 and Figure 5. From the table, it is seen that training with 'compensated threshold' offers improved performance as compared with the experiments with 'equal threshold'. It was found also from the experiments, recognition errors occurred were mostly due to similar postures belonging to different training posture groups, as shown for example in Figure 6.

No. of added postures	1	3	5
Equal Threshold	0.33	0.68	1.72
Compensate Threshold	0.13	0.33	0.51

Table 2. Recognition error rate (%) for different numbers of added postures.

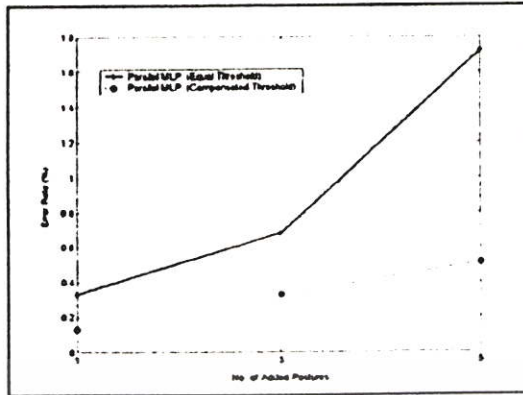


Figure 5. Plot of data shown in Table 2.



Figure 6. Example of a posture pair which caused false classification

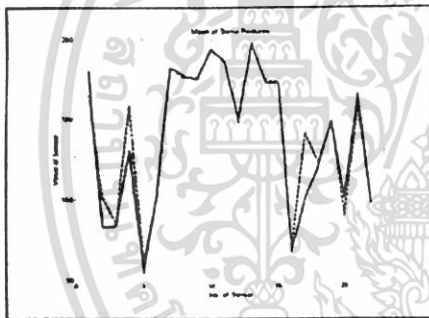


Figure 7. Mean values of the pattern pair of Figure 6.

5. Conclusion

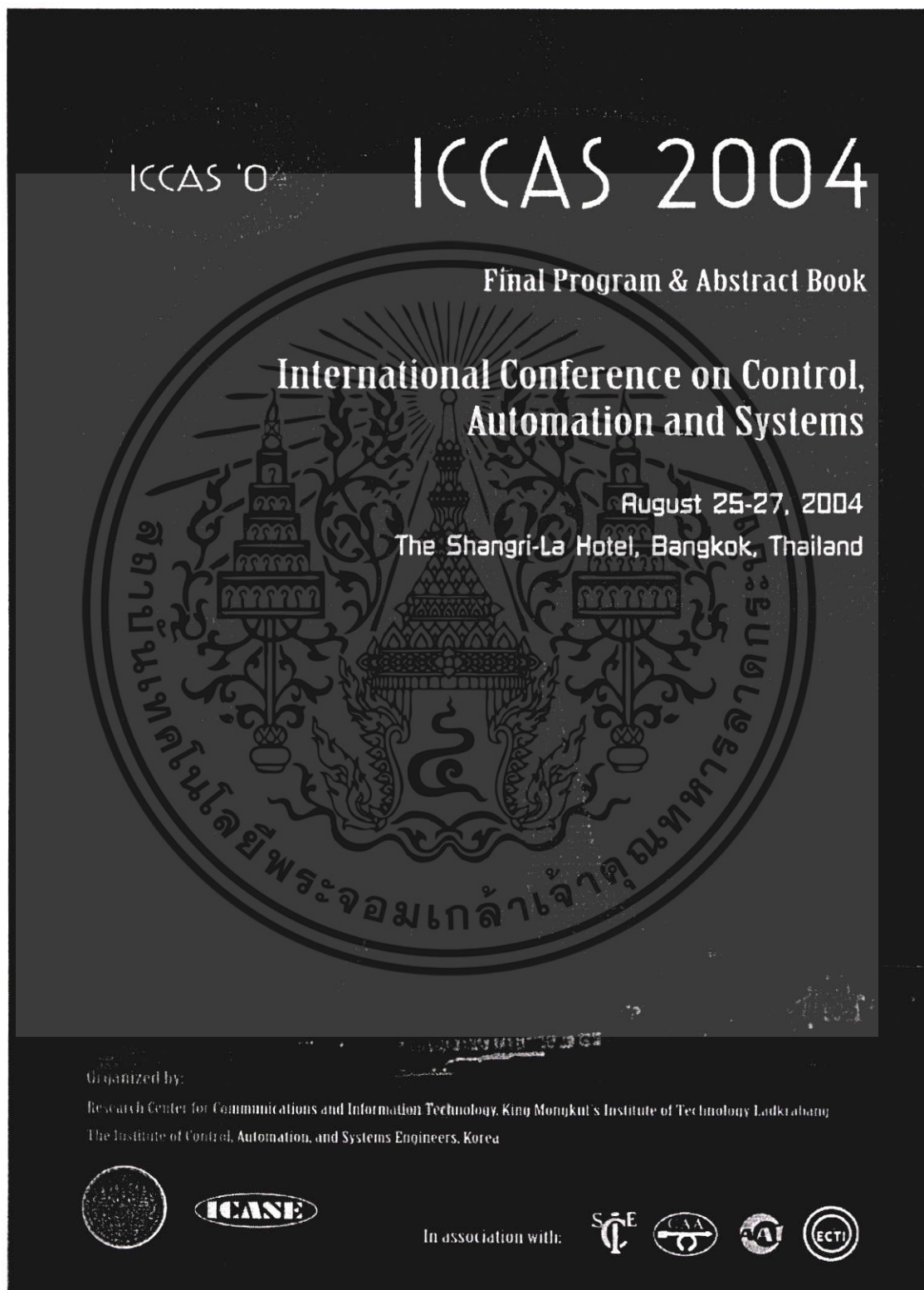
In this paper, the parallel multi-layer Perceptron network has been proposed. The architecture has been motivated by the need to incrementally perform network retraining, when a small number of new patterns are added to the system. The network has been derived by decomposing a standard Bayesian estimator into linear and non-linear terms. Two neural networks with multi-layer and single-layers are used in parallel to model the non-

linear and linear terms, respectively. Experimental results have been included to show the recognition error for the case where a small number of new patterns are added. Cause of recognition error has been identified. Future work includes the re-training strategy which involves only the new patterns, and improves the accuracy result.

References

- [1] Manjula B. Waldron, "Isolated ASL Sign Recognition System for Deaf Persons", IEEE Transactions on Rehabilitation Engineering, Vol. 3, No. 3, September 1995, pp. 261-271.
- [2] S. Sidney Fels and Geoffrey E. Hinton, "Glove-Talk : A Neural Network Interface Between a Data - Glove and a Speech Synthesizer", IEEE Transactions on Neural Network Vol. 4, No. 1, January 1993.
- [3] Philip A. Harling, "Gesture Input using Neural Network", BSc degree, Department of Computer Science University of York Heslington, UK, March 1993.
- [4] Worawit Werapan and Nopporn Chotikakamthorn, "Gesture Segmentation Technique for Thai Sign Language Translation", The Second International Symposium on Communications and Information Technology, Chonburi, Thailand, 23 - 25 October 2002, pp. 483-486.
- [5] James A. Freeman, Simulating Neural Networks with Mathematica, Addison-Wesley Publishing Company, USA, 1994.
- [6] Morton Nadler, Eric P. Smith, Pattern Recognition Engineering, John Wiley & Sons, Inc., Canada, 1993.

2. Kumphol Wasanapongpan and Nopporn Chotikakamthorn. "Artificial Neural Network for Quantitative Posture Classification in Thai Sign Language Translation System" ICCAS 2004, pp. 483 - 486



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Acceptance Letter



2004 International Conference on Control, Automation and Systems (ICCAS 2004)

August 25-27, 2004
The Shangri-La Hotel, Bangkok, Thailand

Home page : <http://www.kmitl.ac.th/iccas04>
<http://www.iccas.org>

■ **General Chair**
Prakit Tangisanon (KMITL, President, Thailand)

■ **General Co-Chair**
Keh Sik Min (ICASE, President, Korea)

■ **Advisory Council**
Taweesak Koranantakul (NECTEC, Thailand)
Hironobu Ono (SICE, Japan)
Han Pu Chen (CAA, China)
Youngil Youn (POSTECH, Korea)
Hidenori Kimura (RIKEN, Japan)
Toshio Wakabayashi (Tokai Univ., Japan)
Monai Krairiksh (KMITL, Thailand)

■ **Organizing Chair**
Tawil Paungma (KMITL, Thailand)

■ **Organizing Co-Chair**
Sangchul Won (POSTECH, Korea)

■ **Conference Program Chair**
Ouen Pinnern (KMITL, Thailand)

■ **Technical Co-Chairs**
Jun-ichi Takada (ITT, Japan)
Ruttikorn Varakulsiripunth (KMITL, Thailand)
Narong Yoothanom (SPU, Thailand)
David Banjerdpongchai (CU, Thailand)
Akachai Sang-In (CMU, Thailand)
Sujate Jantarang (MUT, Thailand)
Waree Kongprawechnon (TU, Thailand)
Sinchai Chinvorarat (KMITNB, Thailand)
Suthee Phoojaruenchanachai (NECTEC, Thailand)
Bhaisai Hoonkeo (SAU, Thailand)

■ **International Relations Co-Chairs**
Jongkol Ngarmwivit (KMITL, Thailand)
Hyun Sik Ahn (Kookmin Univ., Korea)
Yutaka Yamamoto (Kyoto Univ., Japan)
Apinunt Thanachayanont (KMITL, Thailand)
Sunpasit Limnararat (KMITL, Thailand)

■ **Regional Program Co-Chairs**
Joonhong Lim (Hanyang Univ., Korea)
Noriyuki Komine (Tokai Univ., Japan)

■ **Exhibition Co-Chairs**
Werachet Khangern (KMITL, Thailand)
Jin Bao Park (Yonsei Univ., Korea)
Mongkol Mongkolwongroj (KMITL, Thailand)

■ **Publication Co-Chairs**
Nopporn Chotikakamthorn (KMITL, Thailand)
Surapan Airphaiboon (KMITL, Thailand)
Young Il Lee (SNUT, Korea)

■ **Secretariats**
Taworn Benjanarasuth (KMITL, Thailand)
Hyun-Chang Yang (ICASE, Korea)
Vimolluck Thianjew (KMITL, Thailand)

May 21, 2004

Paper No : ABST 145

Title : Artificial Neural Network for Quantitative Posture Classification in a Thai Sign Language Translation System

Authors : Mr. Wasanapongpan Kumphol, Dr. Chotikakamthorn Nopporn

Dear : Mr. Wasanapongpan Kumphol, Dr. Chotikakamthorn Nopporn

I would like to express my gratitude for your continued attention and support for the ICCAS2004. On behalf of the ICCAS 2004 Program Committee, it is a great pleasure to inform you that your paper described above has been accepted for presentation. You are requested to submit the paper and abstract under "Online Submission > Submit Full Paper" in ICCAS04 web site (<http://www.kmitl.ac.th/iccas04>) by **June 21, 2004**. Please follow the guideline as can be found in the 'Author's Kit' section of the ICCAS04 web site, as well as those shown below when preparing the manuscript.

✕ **Guidelines**

1. A manuscript must be submitted as a PDF file. Make sure that the manuscript uses only English characters.
2. A manuscript is limited to 6 single-spaced two-column pages including the abstract, references, figures and tables.

Thank you for your contribution to the ICCAS2004.

We are looking forward to seeing you at The Shangri-La Hotel, Bangkok, Thailand.

Yours Sincerely,

Tawil Paungma

ICCAS 2004 Organizing Chair
<http://www.kmitl.ac.th/iccas04>

ICCAS2004 Organizing Chair Tawil Paungma
Research Center for Communications and Information Technology (ReCCIT)
King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, THAILAND
TEL: +66-27392427-8, FAX: +66-27392429
E-mail: iccas04@kmitl.ac.th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Artificial Neural Network for Quantitative Posture Classification in Thai Sign Language Translation System

Kumphol Wasanapongpan, and Nopporn Chotikakamthorn

Faculty of Information Technology &

Research Center for Communications and Information Technology

King Mongkut's Institute of Technology Ladkrabang, Bangkok Thailand

(E-mail: s3067019@kmitl.ac.th)

Abstract: In this paper, a problem of Thai sign language recognition using a neural network is considered. The paper addresses the problem in classifying certain signs conveying quantitative meaning, e.g., large or small. By treating those signs corresponding to different quantities as derived from different classes, the recognition error rate of the standard multi-layer Perceptron increases if the precision in recognizing different quantities is improved. This is due the fact that, to increase the quantitative recognition precision of those signs, the number of (increasingly similar) classes must also be increased. This leads to an increase in false classification. The problem is due to misinterpreting the amount of quantity the quantitative signs convey. In this paper, instead of treating those signs conveying quantitative attribute of the same quantity type (such as 'size' or 'amount') as derived from different classes, here they are considered instances of the same class. Those signs of the same quantity type are then further divided into different subclasses according to the level of quantity each sign is associated with. By using this two-level classification, false classification among main gesture classes is made independent to the level of precision needed in recognizing different quantitative levels. Moreover, precision of quantitative level classification can be made higher during the recognition phase, as compared to that used in the training phase. A standard multi-layer Perceptron with a back propagation learning algorithm were adapted in the study to implement this two-level classification of quantitative gesture signs. Experimental results obtained using an electronic glove measurement of hand postures are included.

Keywords: Sign Language, Gesture Recognition, Artificial Neural Network, Quantitative Posture

1. INTRODUCTION

Gesture classification is a form of pattern recognition problems. Like other pattern recognition problems, neural networks have also been applied to solve gesture recognition problems. For example, in [1] a neural net was applied for isolated American sign language (ASL) recognition. Their system obtained gesture data from an electronic (data) glove and achieved the accuracy level of 86%. In [2], a gesture-recognition system called Glove-Talk was developed, using a neural network for linking information from a data glove and a speech synthesizer. In [3], a neural network was applied to recognition a small number of hand postures representing English alphabets. In [4], a problem of segmenting continuous Thai signs is considered. The Bayesian estimator is applied in the paper for sign recognition.

One of the most popular neural network architecture is a multi-layer Perceptron. However, the network has a weakness in terms of the difficulty to strictly confine the hidden-layer multi-dimensional space regions of similar patterns, as constructed by the network during the training stage, to be close to one another. As a result, similar patterns may be recognized as totally different patterns. This is likely to occur for patterns that are not included in the training set (or those that are not well represented by the training patterns).

In Thai sign language, there are some hand postures that convey quantitative information. Examples of such signs include those that represent 'size' and 'amount' quantity type. For each quantity type, a sign can describe different levels of quantitative attribute by slightly changing hand posture in a well-defined manner. Figure 1 shows an example of such quantitative hand posture signs.

To apply a standard multi-layer Perceptron to recognize signs in which those quantitative postures are included, hand postures corresponding to the same quantity type may be divided into some number of classes. Each class contains the hand postures that convey the same level of quantitative

attribute. For example, there may be three quantitative classes for the hand postures representing different quantitative levels of 'size' (as may be labeled by 'small', 'medium', 'large'). This approach does not need any modification made to the network structure, and standard back propagation algorithm can be directly applied. However, the approach has two weaknesses. First, the number of classes to be recognized is increased. The higher precision in recognizing different quantitative levels is needed, the higher number of classes is required. This can result in an increase in the complexity of the network. With the fact that the number of such quantitative signs as found in Thai sign language is relatively low, this complexity problem may not be a serious concern.

The second problem with the approach as described is due to similarity among quantitative and non-quantitative hand postures. Firstly, same-quantitative-type hand postures conveying similar quantitative levels are difficult to differentiate due to the similarity of their hand shapes. Secondly, insufficient degree of freedom in a network model, as well as insufficient number of the hand postures used in the training set, can result in some quantitative hand postures being misclassified as non-quantitative postures and vice versa. The latter is due to the difficulty in controlling the Perceptron to strictly confine the hidden-layer multi-dimensional space regions of similar patterns to be close to one another. In any of the two cases, the recognition rate is decreased.



Fig. 1. Example of quantitative hand postures

This paper proposes to use a two-level classification for quantitative hand posture recognition to solve the problems as described. The first level is used to classify those quantitative postures according to the quantity types they represent. The second level is used to identify the posture's quantitative level. The paper is organized as follows. First, a theory and notation of multi-layer Perceptron neural network as applied to Thai sign language recognition are introduced in Section 2. Next, the proposed two-level quantitative posture classification is described in Section 3. Experiment results, obtained from a real measurement using an electronic glove, are provided with some discussion in Section 4. A conclusion remark is given in Section 5.

2. MULTI-LAYER PERCEPTRON FOR SIGN LANGUAGE RECOGNITION

The MLP is a supervised neural network. A general structure of a multi-layer Perceptron neural network is shown in Figure 2.

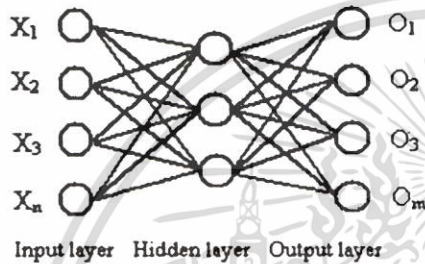


Fig. 2. Standard multilayer perceptron neural network

The MLP use the back-propagation (BP) algorithm, [5-7] for training. The algorithm consists of 2 phases : a feed-forward process and a back-propagation process. For the initial stage, the weights of the network are randomly selected. The learning rate η is pre-set before the learning phase. Given x_i is the vector of n dimension. It is fed to the network input layer. The i^{th} input layer is connected to the j^{th} node of the hidden layer through the weights W_{ij} . $i = 1, 2, \dots, n$, and $j = 1, 2, \dots, s$. The j^{th} hidden layer is connected to m^{th} node of the output layer through the weights Z_{kj} . $k = 1, 2, \dots, m$.

Where

- n = number of input layer node
- s = number of hidden layer node
- m = number of output layer node

The output of the hidden layer is obtained by passing the summation of the multiplication between the inputs and the weights (W), through the activation function, as described by

$$h_j = f(\text{net}_j) \quad (1)$$

$$\text{net}_j = \sum_{i=1}^n W_{ij} x_i \quad (2)$$

where $f(\cdot)$ is the activate function or transfer function. Here, $f(\cdot)$ is chosen as a sigmoidal function given below.

$$f(x) = \frac{1}{1 + \exp(-x)} \quad (3)$$

Next, the output from the hidden layer is fed as an input to the output layer. The output of the output layer is obtained by passing the summation of the multiplication between the hidden output and the weights (Z), through the activation, as described by

$$o_k = f(\text{net}_k) \quad (4)$$

$$\text{net}_k = \sum_{j=1}^s Z_{kj} h_j \quad (5)$$

The output vector (o_k), $o_k = [o_1, o_2, \dots, o_m]$ generated from the feed-forward process is compared with the desire response vector (t_k), $t_k = [t_1, t_2, \dots, t_m]$. The cost function or error (e) used here is a square error function, which is given by

$$e = \frac{1}{2} \sum_{k=1}^m (t_k - o_k)^2 \quad (6)$$

Back-propagation process is then using error to adjust weights as follows:

$$Z_{(j+1)} = \eta \delta_k h_j + Z_{(j)} \quad (7)$$

where

$$\delta_k = o_k (1 - o_k) (t_k - o_k) \quad (8)$$

$$W_{(j+1)} = \eta \beta_j x_i + W_{(j)} \quad (9)$$

where

$$\beta_j = h_j (1 - h_j) \sum_k \delta_k W_{kj} \quad (10)$$

The network as described is here applied to the sign language recognition problem. Only the static and isolated postures are considered here. The problems of static/dynamic segmentation, as well as dynamic gesture classification were discussed elsewhere (see [4], for example). Under this scope, X 's are data as measured from an electronic glove sensors. In this study, a total of 22 sensors were placed on the glove. Thus, the number of input signals $n = 22$. The outputs O correspond to the hand posture classes.

3. TWO-LEVEL METHOD FOR QUANTITATIVE POSTURE CLASSIFICATION

In this section, the proposed two-level method for quantitative hand posture classification is described.

3.1 Network structure

A structure of the multi-layer perceptron neural network, modified to accommodate the two-level classification scheme is shown in Figure 3. From this point, the network is referred as QMLP (Quantitative-classification Multi-Layer Perceptron)

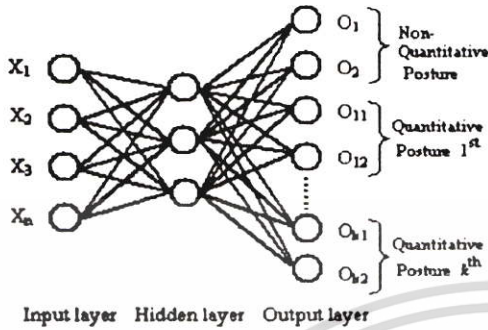


Fig. 3. A multi-layer Perceptron for quantitative posture classification

From Fig. 3, the network outputs can be divided into two categories. The first one, in which there are two output nodes O_1 and O_2 in the figure, corresponds to groups of the non-quantitative posture outputs. The second category is a group of quantitative posture outputs. From the figure, it is noted that for each quantitative posture type, there are two output nodes associated with it. For example, the output nodes $O_{1,1}$ and $O_{1,2}$ belongs to the 1st quantitative posture type (or class). This two-node per class arrangement makes possible the separation between quantitative posture type classification and the quantitative level classification. This point will be clarified later in this section.

3.2 Defining desired responses for supervised training

To train the network in Fig. 3, a desired response for each training input must be defined. Desired response for the non-quantitative and quantitative postures are defined differently as described below.

- For the input data corresponding to an i^{th} non-quantitative posture class, desired response of the output of the i^{th} non-quantitative posture is set to 0.9. Desired responses of other outputs node are set to 0.1.
- For the input data corresponding to an i^{th} quantitative posture class, two desired responses corresponding to the i^{th} quantitative posture are defined as a point on the circle as described in Fig. 4.

From Fig. 4, $d = 0.9$ is the circle radius, and θ is the angular value representing the quantitative level. For example, in this paper, the largest quantitative level for any quantity type is associated with $\theta = 85^\circ$, while the smallest level is associated with $\theta = 5^\circ$. The levels in between the two extremes are

proportionally given angular values between 5 and 85. To simplify labeling the input data with an appropriate quantitative level, however, the whole range of θ may be divided into N angular intervals. Each of those intervals correspond to a particular quantitative level. Thus, the value of N controls the precision of the recognizer in differentiating postures of similar quantitative levels.

With θ being given according to the quantitative level of the training input data, the corresponding desired responses of the associated output pairs ($O_{i,1}$ and $O_{i,2}$ in the figure) are given by

$$O_{i,1} = d \cdot \cos(\theta) \quad (11)$$

$$O_{i,2} = d \cdot \sin(\theta) \quad (12)$$

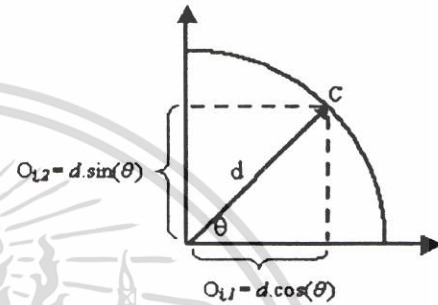


Fig. 4. Mapping between the quantitative level and the desired response pair for quantitative posture data

3.3 Quantity type and quantitative level classification

After training the network using the method for defining the desired responses, as just detailed, the network can be used for hand posture recognition. However, because each quantitative posture class has two output nodes, values of these outputs must be first transformed so that they can be compared against the output nodes of non-quantitative postures. To do so, for arbitrary input data, let's $o_{i,1}$ and $o_{i,2}$ be the output values corresponding to $O_{i,1}$ and $O_{i,2}$ respectively. From these output values, compute the following transformed output values

$$\hat{d}_i = \sqrt{(o_{i,1})^2 + (o_{i,2})^2} \quad (13)$$

$$\hat{\theta}_i = \tan^{-1}(o_{i,2} / o_{i,1}) \quad (14)$$

From these two transformed output values, \hat{d}_i is used to compared against other output values, to obtain the output that yields the highest value. The input data is then recognized as belonging to the class corresponding to the output node with the highest value. In case that the i^{th} quantitative class is the winning class, the value of $\hat{\theta}_i$ is used to obtain the associated quantitative level. By comparing $\hat{\theta}_i$ against the lower and upper bounds (5 and 85 degrees, respectively), a comparative quantitative level of the input data can be identified.

Alternatively, if the whole range of θ is divided into N angular intervals during the training stage, the input data is classified as belonging to one of those N quantitative levels.

4. EXPERIMENT

In the experiment, hand posture data was collected using an electronic glove containing 22 sensors. Five signers participated in the experiment. Each of them was asked to perform a total of 25 different static non-quantitative postures, and 3 static quantitative postures. For each quantitative posture, each signer was asked to perform signs corresponding to 5 different quantitative levels. Therefore, each signer performs 40 signs in total. Each sign was repeatedly performed for 460 times.

A standard multi-layer Perceptron method

When a standard multi-layer Perceptron was used, for training and testing, all 40 signs were first classified into Q different classes, where Q was varied among 4 experiments (2, 3, 5 and 9). However, if Q equal to 9, we were first interpolated each sign's quantitative levels from 5 to 9 levels because we have only 5 different quantitative levels, therefore all signs were 52 signs.

The desired response of 0.9 was given to the output node corresponding to the correct posture class, while the response of 0.1 was given to the other nodes. Number of hidden nodes in the hidden layer is 40, 60 and 80 nodes.

The QMLP method

In the experiment using QMLP method, for the postures of network training, we were first interpolated each sign's quantitative levels from 5 to 17 levels. Afterward, all 17 sign's quantitative levels were classified into Q different classes, where Q was varied among 4 experiments (2, 3, 5 and 9).

However, because each quantitative class require two output nodes, the total number of the output nodes was 31. The number of the hidden nodes was the same as used in the standard multi-layer Perceptron method.

Table 1. Recognition rates of the two methods

Neural Network Type	Accuracy (%)
Standard Method	100
QMLP Method	100

Table 1 compares the recognition rates of the both methods as described. The figures as shown were calculated from the ability of a recognizer in providing correct posture classes for non-quantitative postures, and correct classes of quantity type for quantitative postures. Misinterpreting the quantitative levels is not counted as a recognition error. From the table, both methods give the same result.

Next, accuracy in classifying the quantitative level for quantitative postures is analyzed. The quantitative level classification errors of the two methods are shown in Table 2 - 5. The error rate was computed as a percentage of quantitative level misclassification, weighted by the absolute difference between the correct quantitative level and the one estimated by any of the two recognizers.

Table 2 . Results using quantitative data classified into 2 classes during the training phase.

No. of Recognition Precision Level	No. Hidden Node	Accuracy (%)		
		40	60	80
2		100	100	100

a) Standard MLP Method

No. of Recognition Precision Level	No. Hidden Node	Accuracy (%)		
		40	60	80
2		99.52	99.73	99.60
3		93.64	93.15	93.37
5		91.06	91.20	91.41
9		90.98	91.44	91.36

b) QMLP Method

Table 3 . Results using quantitative data classified into 3 classes during the training phase.

No. of Recognition Precision Level	No. Hidden Node	Accuracy (%)		
		40	60	80
3		100	100	100

a) Standard MLP Method

No. of Recognition Precision Level	No. Hidden Node	Accuracy (%)		
		40	60	80
3		98.46	98.76	98.72
5		97.20	96.96	96.76
9		96.84	96.98	96.92

b) QMLP Method

Table 4. Results using quantitative data classified into 5 classes during the training phase.

No. of Recognition Precision Level	No. Hidden Node	Accuracy (%)		
		40	60	80
5		99.20	99.70	99.59

a) Standard MLP Method

No. of Recognition Precision Level \ No. Hidden Node	Accuracy (%)		
	40	60	80
5	98.30	98.11	98.18
9	96.33	96.56	96.51

b) QMLP Method

Table 5. Results using quantitative data classified into 9 classes during the training phase.

No. of Recognition Precision Level \ No. Hidden Node	Accuracy (%)		
	40	60	80
9	98.84	99.00	98.96

a) Standard MLP Method

No. of Recognition Precision Level \ No. Hidden Node	Accuracy (%)		
	40	60	80
9	98.05	98.32	98.09

b) QMLP Method

From the above tables, it was found that the standard MLP performed slightly better than the QMLP, when the recognition precision level matches the quantitative levels used during the training phase. Unlike the standard method, however, the QMLP method can be used to classify quantitative postures using a higher precision level than that used in the training stage. For example, in Table 3, three times increase in the precision level during the recognition stage, as compare with the training stage, result in only slightly decrease in accuracy levels. Nonetheless, in all experiments, the accuracy was found to be reduced when the recognition precision level was increased.

5. CONCLUSION

In this paper, the quantitative-classification multi-layer Perceptron network has been proposed. The architecture has been motivated by the need to deal with quantitative postures (those that convey quantitative level information), since the standard multi-layer Perceptron inefficient when applied to those posture classes. The paper proposes the use of the two-level classification scheme, applied to the standard multi-layer Perceptron. Details on assigning the desired response as well as the classification decision for the proposed method have been given. Experimental results included show that the propose method offers higher quantitative recognition precision, as compared with the number of quantitative levels used in the training phase. This can greatly simply the network training, and yet provides high precision in quantitative level classification.

REFERENCES

- [1] Manjula B. Waldron, "Isolated ASL Sign Recognition System for Deaf Persons", IEEE Transactions on Rehabilitation Engineering, Vol. 3, No. 3, September 1995, pp. 261-271.
- [2] S. Sidney Fels and Geoffrey E. Hinton, "Glove-Talk : A Neural Network Interface Between a Data - Glove and a Speech Synthesizer", IEEE Transactions on Neural Network Vol. 4, No. 1, January 1993.
- [3] Philip A. Harling, "Gesture Input using Neural Network", BSc degree, Department of Computer Science University of York Heslington, UK, March 1993.
- [4] Worawit Werapan and Nopporn Chotikakamthorn, "Gesture Segmentation Technique for Thai Sign Language Translation", The Second International Symposium on Communications and Information Technology, Chonburi, Thailand, 23 - 25 October 2002, pp. 483-486.
- [5] D.E. Rumelhart et al., "Parallel Distributed Processing : Explorations in the Micro Structure of Cognition Vol. I: Foundations", MIT Press, Cambridge, MA, 1986.
- [6] Rumelhart, D.E. Hinton, G.E. and Williams, R.J. "Learning internal representations by error propagation", In Rumelhart, D.E. and McClelland, J.L. Eds, Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1, pp.318-362, 1986
- [7] Parker, K.L. and Thornbrugh, A.L. "Parallelized back - propagation training and its effectiveness", International Joint Conference on Neural Network, Vol. 2, pp.179-182, Washington, DC, 1990.

ประวัติผู้เขียน

ชื่อผู้เขียน	นายกำพล วาสนาพงศ์พันธุ์
วัน เดือน ปีเกิด	วันที่ 17 ตุลาคม พ.ศ. 2519
วุฒิการศึกษาระดับปริญญาตรี	ครุศาสตรบัณฑิต สาขาคอมพิวเตอร์ศึกษา
สถานที่สำเร็จการศึกษา	สถาบันราชภัฏ อุบลราชธานี
ปีที่สำเร็จการศึกษา	2543
ประสบการณ์การทำงาน	รับราชการ ตำแหน่ง อาจารย์ 1 ระดับ 4 โรงเรียนประชาราษฎร์อุปถัมภ์วิทยา สังกัดกรุงเทพมหานคร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้