การรู้จำตัวอักษรพิมพ์ภาษาไทยโดยการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก

นางสาวสุขวสา พิชิตเดช

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2544
ISBN 974-03-1351-5
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

THAI PRINTED CHARACTER RECOGNITION USING A NEURAL NETWORK ENSEMBLE

Miss Sukwasa Pichitdej

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering in Computer Engineering

Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University

Academic Year 2001
ISBN 974-03-1351-5

หัวข้อวิทยานิพนธ์	การรู้จำตัวอักษรพิมพ์ภาษาไทยโดยการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก
โดย	นางสาวสุขวสา พิชิตเดช
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.สมศักดิ์ ปัญญาแก้ว)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(รองศาสตราจารย์ ดร.สมชาย ประสิทธิ์จูตระกูล)

..... อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล)

..... กรรมการ
(รองศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา)

..... กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.ธนารักษ์ ธีระมั่นคง)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สุขวสา พิชิตเดช : การรู้จำตัวอักษรพิมพ์ภาษาไทยโดยการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก. (THAI PRINTED CHARACTER RECOGNITION USING A NEURAL NETWORK ENSEMBLE) อ. ที่ปรึกษา : ผศ.ดร.บุญเสริม กิจศิริกุล, 84 หน้า. ISBN 974-03-1351-5

วิทยานิพนธ์ฉบับนี้มีวัตถุประสงค์เพื่อศึกษาหาวิธีที่เหมาะสมในการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์กเพื่อใช้ในการรู้จำตัวอักษรพิมพ์ภาษาไทยที่พิมพ์จากเครื่องคอมพิวเตอร์ด้วยรูปแบบตัวอักษรมาตรฐาน กลุ่มก้อนคือกลุ่มของตัวแยกแยะที่ทำหน้าที่แยกแยะประเภทของข้อมูลโดยการรวมผลลัพธ์จากตัวแยะแยะหลายๆตัวเพื่อทำนายผลลัพธ์สุดท้าย ซึ่งตัวแยกแยะที่ใช้ในวิทยานิพนธ์ฉบับนี้คือ นิวรอลเน็ตเวิร์ก

วิทยานิพนธ์ฉบับนี้นำเสนอวิธีการต่างๆ เพื่อใช้ในการสร้างกลุ่มก้อนของนิวรอลเน็ตเวิร์กและรวมผลลัพธ์ของกลุ่มก้อนของนิวรอลเน็ตเวิร์ก ซึ่งวิธีการที่นำเสนอนี้เป็นวิธีการรวมผลลัพธ์แบบถ่วงน้ำหนักที่เหมาะสำหรับตัวแยกแยะที่ให้ความถูกต้องสูง คือไม่มีข้อผิดพลาด(หรือมีน้อยมาก)สำหรับข้อมูลที่สอน ผลการทดลองที่ได้แสดงให้เห็นว่าวิธีการที่นำเสนอสามารถทำงานได้อย่างดีและให้ความถูกต้องสูงกว่าวิธีอื่นๆที่ทดสอบในการทดลอง

วิทยานิพนธ์ฉบับนี้ใช้ภาพตัวอักษรต้นแบบที่ได้จากการพิมพ์ด้วยเครื่องพิมพ์เลเซอร์ที่ความละเอียด 600 จุดต่อนิ้ว แล้วนำเอกสารมาอ่านผ่านเครื่องสแกนเนอร์ที่ความละเอียด 200 จุดต่อนิ้ว ซึ่งประกอบด้วยตัวอักษรรูปแบบ AngsanaUPC, BrowalliaUPC, CordiaUPC, DilleniaUPC, EucrosiaUPC และ FreesiaUPC โดยรูปแบบแต่ละรูปแบบจะมีตัวอักษรทั้งหมด 68 ตัว ซึ่งตัวอักษรแต่ละตัวประกอบด้วยตัวอักษรขนาด 16, 18, 20, 22, 24, 26, 28 และ 36 จุด โดยในการเรียนรู้นั้นใช้ภาพของตัวอักษรจำนวน 3264 ตัวอักษร และในการทดสอบการรู้จำ ใช้ภาพตัวอักษรที่ได้จากการนำตัวอักษรต้นแบบไปถ่ายเอกสารให้จางลงและเข้มขึ้น ได้เป็นข้อมูลทดสอบชุดที่ 1 และ 2 ตามลำดับ รวมจำนวนตัวอักษรที่ใช้ทดสอบทั้งหมดเป็น 6528 ตัวอักษร ได้ผลการรู้จำที่มีความผิดพลาดต่ำสุดคิดเป็นร้อยละ 1.53 สำหรับข้อมูลทดสอบชุดที่ 1 และร้อยละ 1.29 สำหรับข้อมูลทดสอบชุดที่ 2

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต .....
สาขาวิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่ออาจารย์ที่ปรึกษา .....
ปีการศึกษา 2544 ลายมือชื่ออาจารย์ที่ปรึกษาร่วม .....

4270594221 : MAJOR COMPUTER ENGINEERING

KEY WORD: ENSEMBLE OF CLASSIFIERS / THAI PRINTED CHARACTER RECOGNITION

SUKWASA PICHITDEJ : THAI PRINTED CHARACTER RECOGNITION USING A NEURAL NETWORK ENSEMBLE. THESIS ADVISOR : BOONSERM KIJSIRIKUL, Assistant Ph.D., 84 pp.ISBN 974-03-1351-5.

The objective of this thesis is to find a suitable method for using an ensemble of neural networks in recognizing Thai printed characters recognition. An ensemble is a group of classifiers that classifies data by combining the results of these classifiers to make the final classification. The individual classifier used in this thesis is a neural network.

This thesis proposes several methods for constructing the ensemble and combining the result of classifiers. This new combining method is a weighted voting method, which is suitable for accurate classifiers making no (or very little) errors on the training data. The experimental results show that our proposed method performs very well on this task and gives higher accuracy than the existing methods tested in our experiment.

The method is tested on prototypic characters that are generated from a laser printer at 600 dots per inch and then are scanned with a scanner at 200 dots per inch. They are composed of characters in 6 fonts: AngsanaUPC, BrowalliaUPC, CordiaUPC, DilleniaUPC, EucrosiaUPC and FreesiaUPC. Each font consists of 68 characters and each character has 8 sizes: 16, 18, 20, 22, 24, 26, 28 and 36 points. In training process 3264 characters are used and in testing process test characters are obtained by copying the prototypic character images by a photocopy machine with lighter and darker settings. Thus, the total test set is composed of 6528 characters. The minimum error rate of recognition for the lighter images is 1.53% and the minimum error rate of recognition for the darker images is 1.29%.

Department	Computer Engineering	Student's signature .....
Field of study	Computer Engineering	Advisor's signature .....
Academic year	2001	Co-advisor's signature .....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดีด้วยความกรุณาอย่างดียิ่งของ ผู้ช่วยศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งได้ให้คำแนะนำและข้อคิดเห็นต่างๆ ในการทำวิจัยด้วยดีมาตลอด

ขอขอบคุณ นายชาญฤทธิ์ สันตินานาเลิศ และนายธเนศ ศรีวิรุฬห์ชัย ที่ให้คำแนะนำในการใช้โปรแกรม Aspirin /MIGRAINES

ขอขอบคุณ ห้องปฏิบัติการ Machine Intelligence and Knowledge Discovery Laboratory ที่เอื้อเฟื้อสถานที่ในการทำวิจัย

ขอขอบคุณ สมาชิกในห้องปฏิบัติการ Machine Intelligence and Knowledge Discovery Laboratory ที่ให้คำแนะนำในการทำวิจัย

ขอขอบคุณ นายกิตติศักดิ์ เลิศเกียรติพานิชย์ ที่ให้คำแนะนำการเขียนโปรแกรมด้วย Visual C++ 6

สุดท้ายนี้ผู้วิจัยขอขอบพระคุณบิดา มารดา ที่ส่งเสริมการศึกษา สนับสนุน และให้กำลังใจแก่ผู้ทำวิจัยเสมอมา

สุขวสา พิชิตเดช

30 เมษายน 2545

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญภาพ ..... ฌ

สารบัญตาราง ..... ญ

บทที่

1. บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 วัตถุประสงค์ของการวิจัย 2

1.3 ขอบเขตของการวิจัย 2

1.4 ขั้นตอนและวิธีการดำเนินการวิจัย 3

1.5 ประโยชน์ที่คาดว่าจะได้รับ 3

1.6 โครงสร้างของวิทยานิพนธ์ 3

1.7 ผลงานที่ตีพิมพ์จากงานวิจัย 4

2. แนวคิดและทฤษฎีที่เกี่ยวข้อง 5

2.1 การวิเคราะห์ตัวประกอบสำคัญ (Principal Component Analysis) 5

2.2 นิวรอลเน็ตเวิร์กแบบแบคพรอพาเกชัน (Backpropagation Neural Network) 6

2.3 กลุ่มก้อนของตัวแยกแยะ (Ensemble of Classifiers) 10

2.3.1 ขั้นตอนการสร้างกลุ่มก้อนของตัวแยกแยะ 12

2.3.2 วิธีรวมผลลัพธ์ 15

2.3.3 ข้อดีของการใช้กลุ่มก้อนของตัวแยกแยะ 16

3. การออกแบบและพัฒนา 18

3.1 ข้อกำหนดต่างๆของนิวรอลเน็ตเวิร์ก 18

3.2 วิธีที่ใช้ในการสร้างกลุ่มก้อนของนิวรอลเน็ตเวิร์ก 19

3.3 วิธีที่ใช้ในการรวมผลลัพธ์จากกลุ่มก้อนของนิวรอลเน็ตเวิร์ก 20

สารบัญ (ต่อ)

บทที่	หน้า
4. การทดลองและผลการทดลอง	25
4.1 แหล่งที่มาของข้อมูลภาพ	25
4.2 วิธีการทดลอง	26
4.2.1 การสร้างกลุ่มก้อนของนิวรอลเน็ตเวิร์ก	26
4.2.2 การรวมผลลัพธ์จากกลุ่มก้อนของนิวรอลเน็ตเวิร์ก	26
4.3 ผลการทดลอง	28
4.3.1 ทดสอบนิวรอลเน็ตเวิร์ก 1 เน็ตเวิร์ก	28
4.3.2 ทดสอบกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างโดยการแบ่งชุดข้อมูลสอน ออกเป็น 6 ชุดตามรูปแบบของตัวอักษร	29
4.3.3 ทดสอบกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างด้วยวิธี Bagging	30
4.4 วิเคราะห์ผลการทดลอง	39
4.4.1 กลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างโดยการแบ่งชุดข้อมูลสอน ออกเป็น 6 ชุดตามรูปแบบของตัวอักษร	39
4.4.2 กลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างด้วยวิธี Bagging	39
4.5 สรุปผลการทดลอง	44
4.6 ปัญหาและข้อจำกัด	45
5. สรุปผลการวิจัยและข้อเสนอแนะ	47
5.1 สรุปผลการวิจัย	47
5.2 ข้อเสนอแนะ	48
รายการอ้างอิง	50
ภาคผนวก	53
ก. ตัวอักษรต้นแบบรูปแบบต่างๆ	54
ข. ตารางแสดงผลการของตัวอักษรที่ผิดแบ่งตามรูปแบบและขนาด	66
ค. ตารางแสดงผลของตัวอักษรที่ผิดแบ่งตามตัวอักษร	70
ประวัติผู้เขียนวิทยานิพนธ์	84

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญภาพ

ภาพประกอบ หน้า

2.1 นิวรอลเน็ตเวิร์กแบบแบคพรอพาเกชันที่มีชั้นที่ถูกซ่อน 1 ชั้น 7

2.2 โครงสร้างของกลุ่มก้อนของตัวแยกแยะ 10

2.3 แสดงค่าความน่าจะเป็นที่ตัวแยกแยะ I ตัว จาก 21 ตัวจะตอบผิดพร้อมๆกัน 11

2.4 แสดงค่าความน่าจะเป็นที่ตัวแยกแยะ I ตัว จาก 21 ตัวจะตอบผิดพร้อมๆกัน 11

2.5 ตัวอย่างชุดข้อมูลสำหรับสอนชุดใหม่ที่ใช้วิธี cross-validate 12

2.6 รหัสขนาด 15 บิต สำหรับปัญหาที่มี 10 คลาส 14

2.7 แสดงให้เห็นเส้นแบ่งขอบเขตที่ได้จากการรวมกลุ่มก้อนของตัวแยกแยะแยะว่ามี
ความใกล้เคียงกับสมมติฐานจริงมากกว่าสมมติฐานจากตัวแยกแยะเพียงตัวเดียว 17

3.1 ขั้นตอนวิธีของวิธีโหวตแบบถ่วงน้ำหนักโดยใช้คลาส (CWV) 22

3.2 แสดงวิธีการทำงานของวิธี Hierarchy Voting 24

4.1 แสดงค่าเปอร์เซ็นต์ความถูกต้องของข้อมูลทดสอบชุดที่ 1
โดยวิธี MV MVP AV MMWV และCWV 31

4.2 แสดงค่าเปอร์เซ็นต์ความถูกต้องของข้อมูลทดสอบชุดที่ 2
โดยวิธี MV MVP AV MMWV และCWV 31

4.3 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก
โดยวิธี MV, MVP, AV, MMWV และ CWV บน ชุดทดสอบที่ 1
เมื่อเทียบกับการใช้นิวรอลเน็ตเวิร์กเดี่ยว 42

4.4 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก
โดยวิธี MV, MVP, AV, MMWV และ CWV บน ชุดทดสอบที่ 2
เมื่อเทียบกับการใช้นิวรอลเน็ตเวิร์กเดี่ยว 42

สารบัญตาราง

ตาราง หน้า

4.1 ค่าความถูกต้องจากนิวรอลเน็ตเวิร์กเดี่ยว 28

4.2 ผลการทดลองกับชุดทดสอบที่ 1 ที่ได้จากเน็ตเวิร์กที่สร้างโดยการแบ่งชุดข้อมูลสอนตามรูปแบบของตัวอักษร แล้วนำผลลัพธ์มารวมกันด้วยวิธีต่างๆ 29

4.3 ผลการทดลองกับชุดทดสอบที่ 2 ที่ได้จากเน็ตเวิร์กที่สร้างโดยการแบ่งชุดข้อมูลสอนตามรูปแบบของตัวอักษร แล้วนำผลลัพธ์มารวมกันด้วยวิธีต่างๆ 29

4.4 ผลการทดลองกับชุดทดสอบที่ 1 ที่ได้จากเน็ตเวิร์กที่สร้างด้วยวิธี Bagging แล้วรวมผลลัพธ์ด้วยวิธีต่างๆ 30

4.5 ผลการทดลองกับชุดทดสอบที่ 2 ที่ได้จากเน็ตเวิร์กที่สร้างด้วยวิธี Bagging แล้วรวมผลลัพธ์ด้วยวิธีต่างๆ 30

4.6 ความผิดพลาดของกลุ่มก้อนของนิวรอลเน็ตเวิร์กแยกตามตัวอักษร บน ชุดทดสอบที่ 1 32

4.7 ความผิดพลาดของกลุ่มก้อนของนิวรอลเน็ตเวิร์กแยกตามตัวอักษร บน ชุดทดสอบที่ 2 34

4.8 ผลการทดลองกับ ชุดทดสอบที่ 1 และ ชุดทดสอบที่ 2 ที่ได้จากเน็ตเวิร์กที่สร้างด้วยวิธี Bagging แล้วรวมผลลัพธ์ด้วยวิธี HV เมื่อใช้จำนวนกลุ่มต่างๆ 36

4.9 ความผิดพลาดของกลุ่มก้อนของนิวรอลเน็ตเวิร์กแยกตามตัวอักษร บน ชุดทดสอบ เมื่อรวมผลลัพธ์ด้วยวิธี HV 37

4.10 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก บน ชุดทดสอบที่ 1 เมื่อเทียบกับการใช้นิวรอลเน็ตเวิร์กเดี่ยว เมื่อรวมผลลัพธ์ด้วยวิธี MV, MVP และ AV 40

4.11 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก บน ชุดทดสอบที่ 2 เมื่อเทียบกับการใช้นิวรอลเน็ตเวิร์กเดี่ยว เมื่อรวมผลลัพธ์ด้วยวิธี MV, MVP และ AV 40

4.12 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก บน ชุดทดสอบที่ 1 เมื่อเทียบกับการใช้นิวรอลเน็ตเวิร์กเดี่ยว เมื่อรวมผลลัพธ์ด้วยวิธี MMWV และ CWV 41

สารบัญตาราง (ต่อ)

ตาราง หน้า

4.13 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก บน ชุดทดสอบที่ 2 เมื่อเทียบกับการใช้นิวรอลเน็ตเวิร์กเดี่ยว เมื่อรวมผลลัพธ์ด้วยวิธี MMWV และ CWV 41

4.14 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของรวมผลลัพธ์ด้วยวิธี HV บน ชุดทดสอบที่ 1 เมื่อเทียบกับรวมผลลัพธ์ด้วยวิธีอื่นๆ 43

4.15 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของรวมผลลัพธ์ด้วยวิธี HV บน ชุดทดสอบที่ 2 เมื่อเทียบกับรวมผลลัพธ์ด้วยวิธีอื่นๆ 43

ข.1 ความผิดพลาดแยกตามรูปแบบและขนาดที่ได้จากการรวมผลลัพธ์ด้วยวิธีต่างๆ เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 1 66

ข.2 ความผิดพลาดแยกตามรูปแบบและขนาดที่ได้จากการรวมผลลัพธ์ด้วยวิธีต่างๆ เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 2 68

ค.1 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี MV เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 1 70

ค.2 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี MV เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 2 71

ค.3 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี MVP เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 1 72

ค.4 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี MVP เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 2 73

ค.5 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี AV เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 1 74

ค.6 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี AV เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 2 75

ค.7 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี MMWV เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 1 76

สารบัญตาราง (ต่อ)

ตาราง หน้า

ค.8 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี MMWV เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 2 77
ค.9 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี CWV เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 1 78
ค.10 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี CWV เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 2 79
ค.11 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี HV (12 กลุ่ม) เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 1 80
ค.12 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี HV (12 กลุ่ม) เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 2 81
ค.13 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี HV (9 กลุ่ม) เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 1 82
ค.14 ความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี HV (9 กลุ่ม) เมื่อใช้ 3 เน็ตเวิร์ก บน ชุดทดสอบที่ 2 83

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันคอมพิวเตอร์นับเป็นอุปกรณ์ที่มีความสำคัญทั้งในชีวิตประจำวันและในการทำงาน และมีแนวโน้มว่าคอมพิวเตอร์จะมีความสำคัญมากขึ้นในอนาคต เนื่องจากความสามารถในการคำนวณ รวมทั้งความสามารถในการเก็บบันทึกและค้นหาข้อมูล ได้อย่างรวดเร็ว ถูกต้องแม่นยำ ซึ่งความสามารถในด้านเหล่านี้จำเป็นสำหรับหลายๆองค์กร

การเก็บบันทึกข้อมูลในรูปแบบอิเล็กทรอนิกส์นั้นสามารถสนองความต้องการขององค์กรในด้านการประหยัดเนื้อที่ในการเก็บข้อมูล และความรวดเร็วในการค้นหาข้อมูลได้เป็นอย่างดี หากแต่การจัดเก็บข้อมูลข่าวสารต่างๆที่มีอยู่เดิมให้อยู่ในรูปแบบอิเล็กทรอนิกส์ เพื่อที่จะสามารถนำไปใช้ในระบบคอมพิวเตอร์ได้นั้น ต้องเสียเวลาและทรัพยากรอย่างมาก นอกจากนี้การพิมพ์ใหม่อาจทำให้มีข้อผิดพลาด ทำให้ข้อมูลข่าวสารนั้นคลาดเคลื่อนจากข้อมูลข่าวสารเดิม เครื่องสแกนเนอร์จึงถูกพัฒนาขึ้นมาเพื่อแก้ปัญหาข้างต้น

การใช้งานเครื่องสแกนเนอร์ เพื่อใช้ในการเก็บรูปภาพและเอกสารต้นฉบับให้อยู่ในรูปแบบอิเล็กทรอนิกส์นั้น จะทำให้ได้เอกสารที่อยู่ในรูปแบบอิเล็กทรอนิกส์ที่ถูกต้องและรวดเร็วกว่าการพิมพ์เอกสารใหม่ หากแต่การเก็บเอกสารในรูปแบบของรูปภาพนั้นสิ้นเปลืองเนื้อที่ฮาร์ดดิสก์ในการเก็บข้อมูลมากกว่าการเก็บในรูปแบบของตัวอักษรและยังยากต่อการแก้ไขในภายหลัง เนื่องจากไม่สามารถใช้โปรแกรมประมวลผลคำทั่วๆไปได้ โปรแกรมโอซีอาร์ (Optical Character Recognition) จึงถูกพัฒนาขึ้น เพื่อแก้ปัญหาดังกล่าว

โปรแกรมโอซีอาร์เป็นโปรแกรมที่จะอ่านข้อมูลในรูปแบบของรูปภาพแล้วแปลงข้อมูลนั้นให้อยู่ในรูปแบบของตัวอักษร เพื่อการจัดเก็บเป็นแฟ้มข้อมูลแบบตัวอักษร ซึ่งจะทำให้สิ้นเปลืองเนื้อที่ในการเก็บข้อมูลในฮาร์ดดิสก์น้อยกว่าการเก็บเป็นแฟ้มข้อมูลแบบรูปภาพ อย่างไรก็ตามโปรแกรมโอซีอาร์นั้นยังไม่สามารถแปลงข้อมูลมาได้ถูกต้องครบถ้วนทั้งหมด

ปัจจุบันโปรแกรมโอซีอาร์ได้มีการพัฒนาสำหรับหลายๆภาษา โปรแกรมโอซีอาร์ภาษาไทยนั้นได้มีการพัฒนามานานแล้วแต่ยังคงมีข้อผิดพลาดในการรู้จำอยู่ จึงยังไม่เหมาะที่จะนำแฟ้มข้อมูลเอกสารที่ได้จากโปรแกรมโอซีอาร์ภาษาไทยนั้นมาใช้งานได้ เพราะต้องแก้ไขข้อมูลในแฟ้ม

ข้อมูลเอกสารให้ถูกต้องเสียก่อน ซึ่งทำให้เสียเวลาและทรัพยากรเป็นจำนวนมาก ซึ่งบางครั้งการพิมพ์เอกสารที่ต้องการขึ้นมาใหม่อาจใช้เวลาน้อยกว่า

ในวิทยานิพนธ์ฉบับนี้จะมุ่งเน้นในการพัฒนาวิธีการสำหรับการรู้จำตัวอักษรพิมพ์ภาษาไทยโดยจะนำเอางานวิจัยที่มีอยู่เดิมมาเป็นส่วนประกอบ และจะใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์กเพื่อปรับปรุงวิธีการสำหรับการรู้จำตัวอักษรพิมพ์ภาษาไทยให้สามารถทำงานได้ถูกต้องมากขึ้น

1.2 วัตถุประสงค์ของการวิจัย

เพื่อศึกษาหาวิธีที่เหมาะสมในการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์กเพื่อการรู้จำตัวอักษรพิมพ์ภาษาไทย

1.3 ขอบเขตของการวิจัย

ออกแบบและพัฒนาโปรแกรมสำหรับการรู้จำตัวอักษรพิมพ์ภาษาไทยโดยการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก โดยมีความสามารถดังนี้

1. โปรแกรมจะทำงานภายใต้ระบบปฏิบัติการ Microsoft Windows บนเครื่องคอมพิวเตอร์ที่มีข้อกำหนดขั้นต่ำคือ ใช้หน่วยประมวลผลกลางรุ่น Pentium 166 มีหน่วยความจำ 80 เมกกะไบต์ขึ้นไปและเนื้อที่ฮาร์ดดิสก์ที่เพียงพอต่อการทำงาน
2. ภาษาคอมพิวเตอร์ที่ใช้ คือ Visual C++ 6.0
3. แฟ้มข้อมูลรูปภาพที่จะใช้ในการทดสอบจะต้องมีลักษณะดังนี้
   - แฟ้มข้อมูลรูปภาพจะต้องมีสีขาว-ดำ จัดเก็บแบบ BMP 1แฟ้มต่อ1 ตัวอักษร
   - รูปแบบตัวอักษรที่ใช้คือ ตัวอักษรพิมพ์ภาษาไทย แบบ AngsanaUPC, BrowalliaUPC, CordiaUPC, DilleniaUPC, EucrosiaUPC, FreesiaUPC
   - ขนาดของตัวอักษรที่จะใช้คือขนาด 16, 18, 20, 22, 24, 26, 28 และ 36 จุด
   - รูปแบบของตัวอักษรจะต้องไม่มีรูปแบบเอียง, รูปแบบหนา หรือขีดเส้นใต้
   - ตัวอักษรที่ทำการรู้จำมีทั้งหมด 68 ตัวแบ่งออกเป็น

     พยัญชนะ 44 ตัว

     - ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ

     สระและสัญลักษณ์ 20 ตัว

     - ะ า ิ ี ึ ื ุ ู ็ ั ์ เ โ ไ ใ ๆ ฤ ฦ ๆ ฯ

วรรณยุกต์ 4 ตัว ดังนี้

- ่ ้ ๊ ๋

4. ตัวอักษรที่ใช้ทดสอบ คือตัวอักษรที่ได้จากการอ่านผ่านเครื่องสแกนเนอร์ในแต่ละรูปแบบในแต่ละขนาดตัวอักษร จำนวน 68 ตัวอักษรและตัวอักษรที่ได้จากการอ่านผ่านเครื่องสแกนเนอร์และทำการขจัดสัญญาณรบกวนประเภทสัญญาณรบกวนแบบจุดเดี่ยวในแต่ละรูปแบบและในแต่ละขนาดตัวอักษรจำนวน 68 ตัวอักษร (รวมทั้งหมด 6528ตัวอักษร)

1.4 ขั้นตอนและวิธีการดำเนินการวิจัย

1. ศึกษาความเป็นไปได้จากผลงานวิจัยที่ผ่านมา
2. ศึกษาทฤษฎีและเลือกวิธีการที่เหมาะสม
3. ออกแบบและพัฒนาวิธีที่เหมาะสมที่สุดสำหรับการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์กในการรู้จำตัวอักษรพิมพ์ภาษาไทย
4. เตรียมข้อมูลที่จะใช้
5. พัฒนาวิธีการสำหรับการรู้จำตัวอักษรพิมพ์ภาษาไทยตามที่ได้ออกแบบไว้
6. ทดสอบและปรับปรุง
7. ประเมินผลโดยการเปรียบเทียบกับผลการรู้จำตัวอักษรพิมพ์ภาษาไทยโดยวิธีอื่นๆ
8. สรุปผลการทำงาน และจัดทำรายงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้วิธีที่เหมาะสมในการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์กเพื่อการรู้จำตัวอักษรพิมพ์ภาษาไทย
2. เป็นแนวทางในการพัฒนาการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก เพื่อเพิ่มประสิทธิภาพของนิวรอลเน็ตเวิร์ก

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

1.6 โครงสร้างของวิทยานิพนธ์

วิทยานิพนธ์นี้แบ่งเนื้อหาออกเป็น 5 บท บทที่ 1 เป็นบทนำซึ่งกล่าวถึงที่มาของปัญหาและวัตถุประสงค์ของงานวิจัย บทที่ 2 สรุปแนวคิดและทฤษฎีที่เกี่ยวข้อง บทที่ 3 อธิบายขั้นตอนวิธีการที่นำเสนอ บทที่ 4 เสนอการทดลองและผลการทดลองของขั้นตอนวิธีที่นำเสนอ และบทที่ 5 เป็นบทที่สรุปผลการวิจัยและข้อเสนอแนะ

1.7 ผลงานที่ตีพิมพ์จากงานวิจัย

ส่วนหนึ่งของวิทยานิพนธ์นี้ได้ตีพิมพ์เป็นบทความทางวิชาการ ในหัวข้อ "กลุ่มก้อนของนิวรอลเน็ตเวิร์กเพื่อการรู้จำตัวพิมพ์อักษรไทย" โดย สุขวสา พิชิตเดช และ บุญเสริม กิจศิริกุล ในงานประชุมวิชาการ "The Fifth National Computer Science and Engineering Conference (NCSEC'2001)" ซึ่งจัดโดยภาควิชาวิทยาศาสตร์คอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่ ณ โรงแรมโลตัสปางสวนแก้ว จังหวัดเชียงใหม่ ในวันที่ 7-9 พฤศจิกายน 2544

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

แนวคิดและทฤษฎีที่เกี่ยวข้อง

ในงานวิจัยนี้เราจะใช้หลักการของกลุ่มก้อนของตัวแยกแยะ (Ensemble of Classifiers) เพื่อการรู้จำตัวพิมพ์อักษรไทย กลุ่มก้อนคือกลุ่มของตัวแยกแยะที่ทำหน้าที่แยกประเภทของข้อมูลโดยการรวมผลลัพธ์จากตัวแยกแยะหลายๆตัวเพื่อทำนายผลลัพธ์สุดท้าย ซึ่งในงานวิจัยนี้จะมีหลักการสำคัญที่ใช้อยู่ 3 หลักการ คือ การวิเคราะห์ตัวประกอบสำคัญ การแยกแยะตัวอักษรโดยใช้แบคพรอพาเกชันนิวรอลเน็ตเวิร์ก และการหาผลลัพธ์โดยใช้กลุ่มก้อนของตัวแยกแยะ

2.1 การวิเคราะห์ตัวประกอบสำคัญ (Principal Component Analysis)

ในการรู้จำตัวพิมพ์อักษรไทยนั้น เราต้องจัดการกับข้อมูลภาพที่มีขนาดค่อนข้างใหญ่ ทำให้ความเร็วของการประมวลผลช้า ในงานวิจัยนี้เราจึงเลือกใช้เทคนิคการวิเคราะห์ตัวประกอบสำคัญ (Principal Component Analysis) [1] เพื่อลดขนาดของภาพและเพิ่มประสิทธิภาพของการรู้จำ การวิเคราะห์ตัวประกอบสำคัญทำได้โดยการหาเวกเตอร์ค่าเฉลี่ย (mean vector) – M_X และ เมตริกซ์โคแวเรียนซ์ (covariance matrix) – C_X สมมติให้ภาพของตัวอักษรที่ต้องการรู้จำมีความกว้าง a1 จุดและความยาว a2 จุด เราแทนแต่ละภาพด้วยเวกเตอร์ X มีขนาดเท่ากับ N (N = a1×a2) โดยที่ N คือจำนวนจุดภาพทั้งหมด (ในงานวิจัยนี้ เวกเตอร์ X มีขนาดเท่ากับ 32×32=1,024) เวกเตอร์ค่าเฉลี่ย M_X และเมตริกซ์โคแวเรียนซ์ C_X ที่มีขนาด N×N คำนวณได้ตามสมการ 2.1 และ 2.2 ตามลำดับ

$$M_x = \frac{1}{T}\sum_{i=1}^{T} X_i \qquad (2.1)$$

$$C_x = \frac{1}{T}\sum_{k=1}^{T} X_k X_k^T - M_x M_x^T \qquad (2.2)$$

โดยที่ T เป็นจำนวนเวกเตอร์ของภาพตัวอย่างทั้งหมดที่ใช้ในการเรียนรู้

จากเมตริกซ์โคแวเรียนซ์ C_X ที่ได้ นำไปหาเมตริกซ์ไอเกนเวกเตอร์ (eigenvectors matrix) – A (ขนาด N×N) ซึ่งการคำนวณเมตริกซ์ไอเกนเวกเตอร์นั้น สามารถทำได้โดยการหาค่าไอเกน (eigen value - λ_i) และไอเกนเวกเตอร์ (eigen vector - e_i) ตามสมการ 2.3

$$C_x e_i = \lambda_i e_i \qquad \text{โดยที่ } i = 1,2,\ldots,N \qquad (2.3)$$

จากนั้นนำไอเกนเวกเตอร์ที่ได้มาสร้างเมตริกซ์ A โดยการนำเอาไอเกนเวกเตอร์ซึ่งได้จากค่าไอเกนที่มีค่ามากสุดเก็บไว้ที่แถวแรก และนำไอเกนเวกเตอร์ซึ่งได้จากค่าไอเกนค่าถัดไปเก็บไว้ที่แถวถัดไปตามลำดับ ดังนั้นแถวสุดท้ายจะเก็บไอเกนเวกเตอร์ซึ่งได้จากค่าไอเกนที่มีค่าที่น้อยที่สุด เมตริกซ์ A ที่ได้จะถูกนำไปใช้สำหรับการแปลงเวกเตอร์ X ซึ่งเป็นเวกเตอร์ที่ได้จากรูปภาพใด ๆ เป็นเวกเตอร์ Y (ขนาด N) ตามสมการที่ 2.4

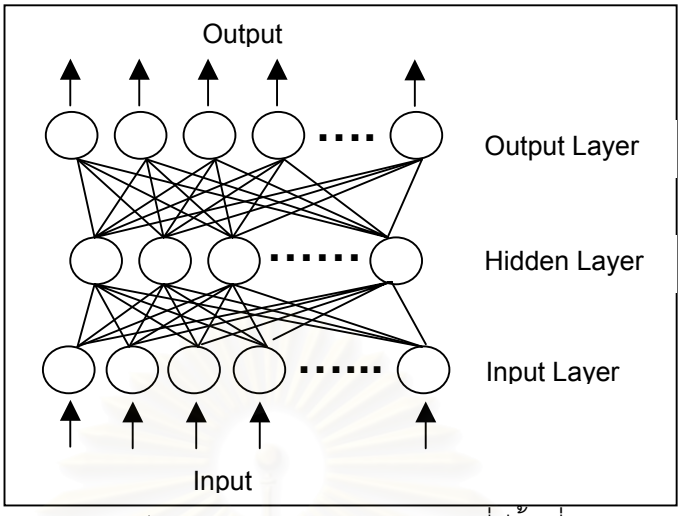
$$Y = A(X - M_X) \tag{2.4}$$

เพื่อลดขนาดของข้อมูล เราจะไม่ใช้ไอเกนเวกเตอร์ทุกตัว แต่จะเลือกใช้เฉพาะไอเกนเวกเตอร์เพียง K ตัวแรก ซึ่งได้จากค่าไอเกนที่มากที่สุด K ตัวแรก ดังนั้นเราจะได้เมตริกซ์ A ซึ่งมีขนาด K×N และเวกเตอร์ Y ที่ได้จะมีขนาด K เนื่องจากค่าไอเกนมีคุณสมบัติที่ว่า ค่าของมันจะลดลงเมื่อลำดับมีค่าสูงขึ้น ดังนั้นเราจะใช้เพียงแค่ค่าไอเกนแรกๆที่มีค่าสูงและจะทิ้งตัวหลังๆที่มีค่าน้อยๆซึ่งถือว่าไม่มีความสำคัญมากนัก ด้วยวิธีการนี้ทำให้เราสามารถลดขนาดข้อมูลจาก N มาเป็น K ได้ จากการทดลองพบว่าค่า K ที่เหมาะสมคือ 128 (จากค่า N=1,024) เมื่อเราได้ข้อมูลของภาพที่ผ่านการวิเคราะห์ตัวประกอบสำคัญแล้ว เราจะได้เวกเตอร์ขนาด K ซึ่งเรียกว่าเวกเตอร์รูปแบบมาเป็นอินพุตสำหรับนิวรอลเน็ตเวิร์กเพื่อทำการรู้จำตัวอักษรต่อไป

2.2 **นิวรอลเน็ตเวิร์กแบบแบคพรอพาเกชัน** (Backpropagation Neural Network)

เป็นขั้นตอนที่นำเอาเวกเตอร์รูปแบบของตัวอักษรที่ได้มาพิจารณาและแยกแยะตัวอักษร (classification) โดยในงานวิจัยนี้ใช้นิวรอลเน็ตเวิร์กแบบแบคพรอพาเกชัน ซึ่งเป็น Feed Forward Network คือในขั้นตอนการทำงานจะไม่มีการป้อนผลลัพธ์ที่ได้ในแต่ละโนดย้อนกลับไปยังโนดที่ส่งข้อมูลมาให้ (แต่ในขั้นตอนการเรียนรู้จะมีการปรับปรุงค่าน้ำหนักของแต่ละโนดเมื่อตอบคำตอบผิด โดยส่งข้อมูลย้อนกลับไปจึงเรียกว่า Backpropagation) ค่าน้ำหนักที่ได้จะเป็นค่าน้ำหนักที่ทำให้ค่าผิดพลาดกำลังสองเฉลี่ย (Mean Square Error) มีค่าต่ำสุด โครงสร้างของนิวรอลเน็ตเวิร์กแบบแบคพรอพาเกชันประกอบด้วย ชั้นอินพุต (input layer) ชั้นซ่อน(hidden layer) ซึ่งอาจมีได้มากกว่า 1 ชั้น และชั้นเอาต์พุต (output layer) ซึ่งแต่ละโนดจะมีการเชื่อมต่อกันแบบเชื่อมต่อทั้งหมด (Fully Connected) ซึ่งแสดงให้เห็นตามรูปที่ 2.1

ในงานวิจัยนี้เราจะให้ชั้นอินพุตของเน็ตเวิร์กประกอบด้วยโนดจำนวน K โนด และในชั้นซ่อนจะใช้จำนวนโนดเท่ากับ 128 โนด ซึ่งเป็นค่าที่ได้จากงานวิจัยของธเนศ [2] (เป็นค่าที่ให้ผลการรู้จำสูงและใช้จำนวนไม่มากเกินไป) ชั้นเอาต์พุตจะมีจำนวนโนดเท่ากับจำนวนตัวอักษรในภาษาไทยที่ใช้ทดสอบซึ่งเท่ากับ 68 โนด



รูปที่ 2.1 นิวรอลเน็ตเวิร์กแบบแบคพรอพาเกชันที่มีชั้นที่ถูกซ่อน 1 ชั้น

<u>การทำงานของนิวรอลเน็ตเวิร์กแบบแบคพรอพาเกชันในขั้นตอนการเรียนรู้</u>

ก่อนที่จะนำนิวรอลเน็ตเวิร์กไปใช้งานในการรู้จำนั้น จะต้องสอนนิวรอลเน็ตเวิร์ก (train) ให้สามารถหาความสัมพันธ์ของ อินพุต และ เอาต์พุต ให้ได้ถูกต้อง หรือผิดพลาดน้อยที่สุดเสียก่อน ซึ่งเรียกว่าขั้นตอนการเรียนรู้ (learning mode, training mode) และมีวิธีการทำงานดังนี้ [3]

- กำหนดค่าเริ่มต้นแก่ค่าน้ำหนัก (weight) และ ค่า bias โดยการสุ่มตัวเลขที่มีค่าน้อยๆ เช่นระหว่าง –0.05 ถึง 0.05
- ส่งข้อมูลที่จะสอนนิวรอลเน็ตเวิร์ก เข้าสู่อินพุตโนด แล้วดำเนินการเหมือนในขั้นตอนการรู้จำ คือ คำนวณค่าผลลัพธ์ของแต่ละโนด (ยกเว้น อินพุตโนด) ตามสมการที่ 2.5 และ 2.6

$$O_j = F(\theta_j + \sum_{i=1}^{n} w_{ji}\ o_i) \qquad (2.5)$$

$$F(a) = \frac{1}{1+e^{-a}} \qquad (2.6)$$

เมื่อ O_j คือผลลัพธ์ของโนดที่ j

θ_j คือค่า bias ของโนดที่ j

w_{ji} คือค่าน้ำหนักของโนดที่ i ซึ่งส่งข้อมูล o_i มาให้โนดที่ j

o_i คือผลลัพธ์ของโนด ที่ i

$F(a)$ คือ ฟังก์ชันกระตุ้น (activation function) ซึ่งใช้ฟังก์ชันซิกมอยด์(sigmoid function)

- ปรับค่าน้ำหนัก

เริ่มต้นที่เอาต์พุตโนด แล้วค่อยทำย้อนกลับไปยังฮิดเดนโนด ในชั้นซ่อน โดยคำนวณค่านำหนักใหม่ (W_{ji}(t+1)) ได้ตามสมการที่ 2.7

$$W_{ji}(t+1) = W_{ji}(t) + \Delta W_{ji} \quad (2.7)$$

เมื่อ W_{ji}(t) คือค่าน้ำหนักของโนด ที่ i ซึ่งเชื่อมกับโนดที่ j เมื่อเวลาที่ t (หรือ iteration ที่ t)

ΔW_{ji} คือค่าน้ำหนักที่ต้องการปรับ

ค่าน้ำหนักที่ต้องการปรับคำนวณได้จากสมการที่ 2.8

$$\Delta W_{ji} = \eta \delta_j O_i \quad (2.8)$$

เมื่อ η คือค่า Learning Rate ($0 < \eta < 1$)

δ_j คือค่าความผิดพลาดของผลลัพธ์ของโนด ที่ j

แต่ในกรณีที่ต้องการที่จะให้นิวรอลเน็ตเวิร์กนั้นลู่เข้า (convergence) เร็วขึ้น ก็สามารถ ทำได้โดยเพิ่มค่า inertia หรือ momentum ตามสมการที่ 2.9

$$W_{ji}(t+1) = W_{ji}(t) + \eta \delta_j O_i + \alpha(W_{ji}(t) - W_{ji}(t-1)) \quad (2.9)$$

เมื่อ α คือค่า Inertia หรือ momentum ($0 < \alpha < 1$)

ค่าความผิดพลาดของผลลัพธ์หาได้ดังนี้

- หากเป็นเอาต์พุตโนดใช้สมการที่ 2.10

$$\delta_j = O_j(1 - O_j)(T_j - O_j) \quad (2.10)$$

เมื่อ T_j คือผลลัพธ์ที่ต้องการได้ (target) จากเอาต์พุตโนดที่ j

O_j คือผลลัพธ์ที่คำนวณได้ (actual) จากเอาต์พุตโนดที่ j

- หากเป็นฮิดเดนโนด ใช้สมการที่ 2.11

$$\delta_j = O_j(1 - O_j)\sum_k \delta_k W_{kj} \quad (2.11)$$

เมื่อ δ_k คือค่าความผิดพลาดของผลลัพธ์ของโนดที่ k (ซึ่งเชื่อมต่อจากโนดที่ j)

W_{kj} คือค่าน้ำหนักของโนดที่ j ซึ่งเชื่อมกับโนดที่ k

ทำการปรับค่าน้ำหนักในโนดอื่นจนครบ แล้วทำการสอนนิวรอลเน็ตเวิร์กในตัวอย่างถัดไป (iteration ถัดไป) จนกระทั่งเน็ตเวิร์กนั้นลู่เข้า

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

เมื่อทำการเรียนรู้ของนิวรอลเน็ตเวิร์กเสร็จสิ้นลง จะเก็บค่าน้ำหนักและค่า bias ของทุกๆ โนด เพื่อนำไปใช้ในการรู้จำต่อไป

<u>การทำงานของนิวรอลเน็ตเวิร์กแบบแบคพรอพาเกชันในขั้นตอนการรู้จำ</u>

ในขั้นตอนการรู้จำ (recognition mode) จะมีการรับข้อมูล (สมาชิกของเวกเตอร์รูปแบบ) เข้าทางอินพุตโนด (ในชั้นอินพุต) จากนั้นแต่ละอินพุตโนดจะส่งข้อมูลไปยังทุกๆ ฮิดเดนโนด (ในชั้นซ่อน) แล้วแต่ละฮิดเดนโนด จะคำนวณค่าที่ได้รับมา ตามสมการ 2.12 และ 2.13

$$O_j = F(\theta_j + \sum_{i=1}^{n} w_{ji}\ o_i) \tag{2.12}$$

$$F(a) = \frac{1}{1+e^{-a}} \tag{2.13}$$

เมื่อ O_j คือผลลัพธ์ของโนดที่ j

θ_j คือค่า bias ของโนดที่ j

w_{ji} คือค่าน้ำหนักของโนดที่ i ซึ่งส่งข้อมูล o_i มาให้โนดที่ j

o_i คือผลลัพธ์ของโนดที่ i

$F(a)$ คือ ฟังก์ชันกระตุ้น (activation function) ซึ่งใช้ฟังก์ชันซิกมอยด์(sigmoid function)

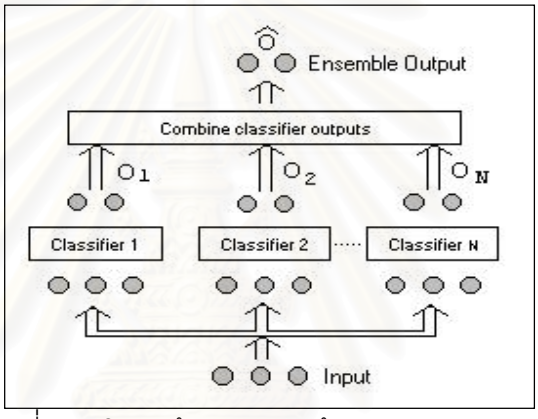
จากนั้นฮิดเดนโนดจะส่งผลลัพธ์ของตัวเองต่อไปยังทุกๆเอาต์พุตโนด (ในชั้นเอาต์พุต) ซึ่งแต่ละเอาต์พุตโนด ก็จะคำนวณค่าที่ได้รับมาตามสมการข้างบน แล้วตอบผลลัพธ์ของโนดตัวเอง ซึ่งเราสามารถพิจารณาผลลัพธ์ของทุกๆเอาต์พุตโนด แล้วหาผลลัพธ์สุดท้ายได้ เช่น ถ้าผลลัพธ์ของ เอาต์พุตโนดที่ 1 มีค่าเท่ากับ 1 และผลลัพธ์ของเอาต์พุตโนดตัวที่เหลือ เท่ากับ 0 ทั้งหมด แสดงว่า ข้อมูลที่ได้รับคือ ข้อมูลของตัวอักษร ก เป็นต้น

หัวข้อต่อไปจะกล่าวถึงการนำกลุ่มก้อนของตัวแยกแยะมาใช้เพื่อช่วยเพิ่มความถูกต้องของการรู้จำ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

2.3 กลุ่มก้อนของตัวแยกแยะ (Ensemble of Classifiers)

การใช้กลุ่มก้อนของตัวแยกแยะ (Ensemble of classifier) เพื่อหาคำตอบของปัญหานั้นสามารถทำได้หลายวิธี โดยปกติมักจะใช้การรวมคำตอบของตัวแยกแยะหลายๆตัวด้วยวิธีใดวิธีหนึ่ง โดยมากมักจะเป็นการรวมแบบถ่วงน้ำหนัก (weighted) หรือไม่ถ่วงน้ำหนัก (unweighted) รูปที่ 2.2 แสดงให้เห็นถึงโครงสร้างของกลุ่มก้อนของตัวแยกแยะ ซึ่งประกอบด้วยตัวแยกแยะหลายๆตัว (Classifier1, Classifier2,…, ClassifierN) ทำหน้าที่รับอินพุตและคำนวณผลลัพธ์ของแต่ละตัว (O_1, O_2, …, O_N) จากนั้นผลลัพธ์โดยรวมของกลุ่มก้อน (Ensemble Output - Ô) หาได้จากการรวมผลลัพธ์ย่อยๆของตัวแยกแยะแต่ละตัวด้วยวิธีการใดวิธีการหนึ่ง

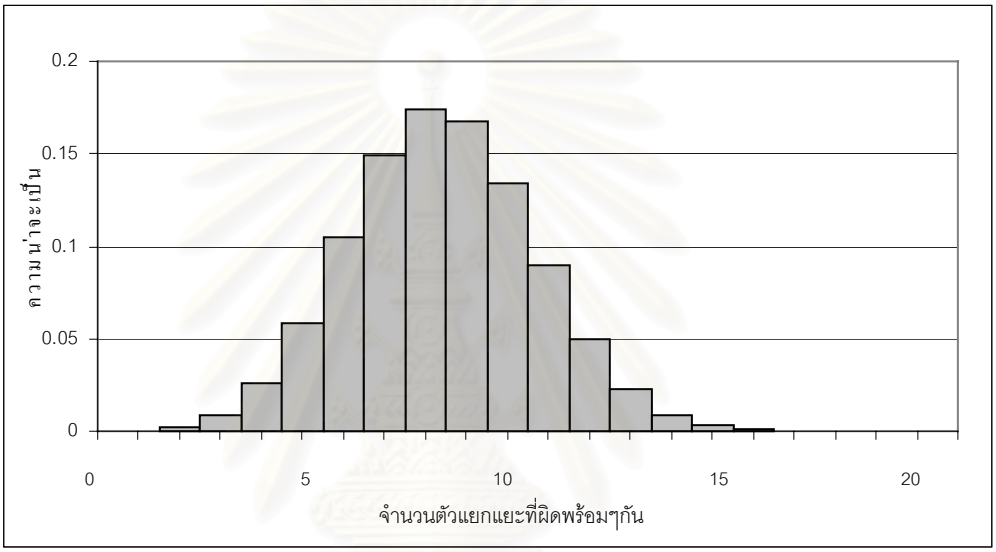


รูปที่ 2.2 โครงสร้างของกลุ่มก้อนของตัวแยกแยะ

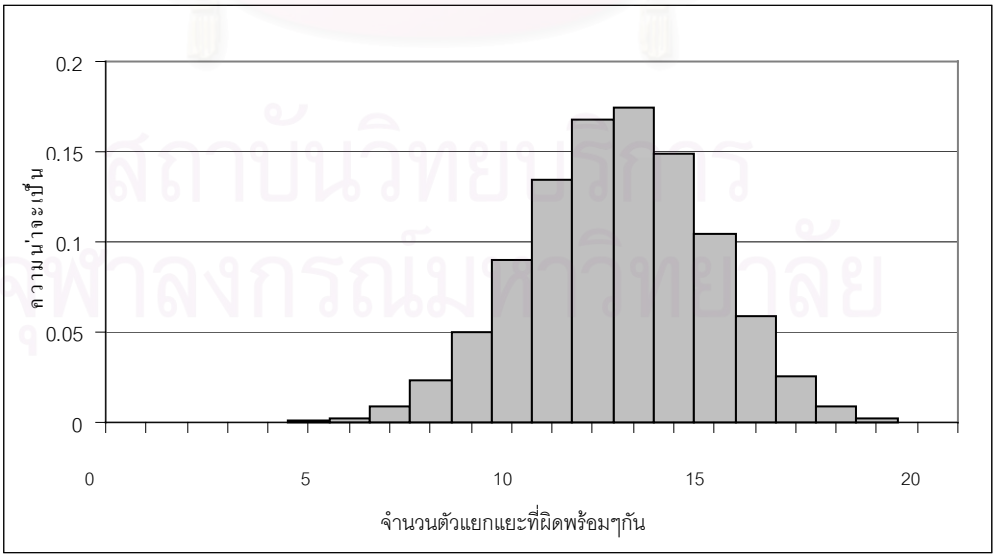
จากการศึกษาทั้งในทางทฤษฎี [4] และทางปฏิบัติ [5] แสดงให้เห็นว่ากลุ่มก้อนของตัวแยกแยะที่ดีจะต้องประกอบด้วยตัวแยกแยะที่มีความถูกต้องสูงและยังมีความหลากหลายในการตอบคำตอบที่ผิด Tumer และ Ghosh [6] ได้แสดงให้เห็นว่าค่าความผิดพลาดของกลุ่มก้อนจะลดลงเป็นศูนย์ได้หากความผิดพลาดจากตัวแยกแยะแต่ละตัวเป็นอิสระจากกัน และจำนวนของตัวแยกแยะเพิ่มขึ้นสู่อนันต์ ในทางตรงกันข้ามถ้าตัวแยกแยะแต่ละตัวมีความผิดพลาดซ้ำๆกัน การรวมผลของตัวแยกแยะที่เหมือนๆกันก็จะไม่ก่อให้เกิดผลดีแต่ประการใด

อย่างไรก็ตามถ้าค่าความผิดพลาดของตัวแยกแยะแต่ละตัวสูงเกินไป การใช้กลุ่มก้อนของตัวแยกแยะในการหาคำตอบของปัญหาก็จะไม่ใช่วิธีที่ดี รูปที่ 2.3 แสดงค่าความผิดพลาดของกลุ่มก้อนที่ประกอบด้วยตัวแยกแยะ 21 ตัว ซึ่งแต่ละตัวมีค่าความผิดพลาดเท่ากับ 0.4 ค่าของพื้นที่ใต้ส่วนโค้งเมื่อตัวแยกแยะตอบผิดพร้อมกันตั้งแต่ 11 ตัวขึ้นไป คือค่าความน่าจะเป็นที่กลุ่มก้อนของตัวแยกแยะจะตอบผิดคิดเป็น 0.1744 จะเห็นได้ชัดว่าค่าความผิดพลาดของกลุ่มก้อนของตัวแยกแยะจะน้อยกว่าค่าความผิดพลาดของตัวแยกแยะเพียงตัวเดียวมาก เพื่อให้เห็นชัดว่าจะ

เกิดอะไรขึ้นถ้าค่าความผิดพลาดของตัวแยกแยะแต่ละตัวมีค่ามากกว่า 0.5 รูปที่ 2.4 แสดงให้เห็นว่า เมื่อตัวแยกแยะแต่ละตัวมีความน่าจะเป็นที่จะตอบผิดเท่ากับ 0.6 พื้นที่ใต้ส่วนโค้งเมื่อตัวแยกแยะตอบผิดพร้อมๆกันตั้งแต่ 11 ตัวขึ้นไป คือค่าความน่าจะเป็นที่กลุ่มก้อนของตัวแยกแยะจะตอบผิดคิดเป็น 0.8256 ดังนั้นกลุ่มก้อนของตัวแยกแยะจะมีความถูกต้องมากกว่าตัวแยกแยะเดี่ยวๆก็ต่อเมื่อตัวแยกแยะแต่ละตัวมีค่าความผิดพลาดน้อยกว่า 0.5



รูปที่ 2.3 แสดงค่าความน่าจะเป็นที่ตัวแยกแยะ I ตัว จาก 21 ตัวจะตอบผิดพร้อมๆกัน



รูปที่ 2.4 แสดงค่าความน่าจะเป็นที่ตัวแยกแยะ I ตัว จาก 21 ตัวจะตอบผิดพร้อมๆกัน

ในการที่จะใช้งานกลุ่มก้อนของตัวแยกแยะนั้น มีขั้นตอนหลักที่ต้องทำอยู่ 2 ขั้นตอน คือ ขั้นตอนการสร้างกลุ่มก้อนของตัวแยกแยะ และขั้นตอนการรวมผลลัพธ์ของตัวแยกแยะ ซึ่งจะกล่าวต่อไปดังด้านล่างนี้

2.3.1 ขั้นตอนการสร้างกลุ่มก้อนของตัวแยกแยะ

วิธีสร้างกลุ่มก้อนของตัวแยกแยะมีหลายวิธี บางวิธีเป็นวิธีที่ใช้ได้ทั่วไป บางวิธีเป็นวิธีที่ใช้เฉพาะสำหรับปัญหานั้นๆ

(1) วิธีการชักตัวอย่างย่อยจากข้อมูลที่ใช้สอน (Subsampling the training example)

Bagging (**B**ootstrap **Agg**regating) โดย Breiman [7] เป็นวิธีสร้างตัวแยกแยะหลายๆตัวที่แตกต่างกันและใช้ตัวแยกแยะเหล่านั้นเพื่อหาคำตอบ วิธีนี้จะสร้างชุดข้อมูลสอนชุดใหม่ (new training data) จากชุดข้อมูลสอนเดิม โดยการสุ่มเอาข้อมูลจากชุดข้อมูลสอนเดิมมาสร้างชุดข้อมูลสอนใหม่หลายๆชุด โดยให้จำนวนของข้อมูลในแต่ละชุดเท่าเดิม (อาจมีข้อมูลบางตัวซ้ำหลายครั้งได้) ซึ่งชุดข้อมูลสอนใหม่แต่ละชุดนี้จะถูกนำไปใช้สร้างตัวแยกแยะตัวใหม่ทีละตัว วิธี Bagging เหมาะสำหรับนิวรอลเน็ตเวิร์กและตัวแยกแยะอื่นๆที่การเปลี่ยนแปลงเพียงเล็กน้อยของชุดข้อมูลสอน ทำให้เกิดการเปลี่ยนแปลงอย่างมากของคำตอบที่ได้

Cross-validate โดย Parmanto, Munro, และ Doyle [8] เป็นวิธีสร้างชุดข้อมูลสำหรับสอนใหม่ โดยการแบ่งชุดข้อมูลสำหรับสอนออกเป็นกลุ่มย่อยๆ แล้วสร้างชุดข้อมูลสำหรับสอนชุดใหม่โดยการตัดกลุ่มย่อยที่แบ่งไว้ออกกลุ่มหนึ่ง รูปที่ 2.5 แสดงให้เห็นชุดข้อมูลสำหรับสอนที่ถูกแบ่งออกเป็น 5 กลุ่มที่ไม่เกี่ยวข้องกัน หลังจากนั้นชุดข้อมูลสำหรับสอนใหม่จะถูกสร้างโดยการไม่รวมกลุ่มย่อยหนึ่งกลุ่มจาก 5 กลุ่ม

1	2	3	4	5	Original
2	3	4	5		Training set 1
1	3	4	5		Training set 2
1	2	4	5		Training set 3
1	2	3	5		Training set 4
1	2	3	4		Training set 5

รูปที่ 2.5 ตัวอย่างชุดข้อมูลสำหรับสอนชุดใหม่ที่ใช้วิธี cross-validate

AdaBoost (**Ada**ptive **Boost**ing) โดย Freund และ Schapire [9,10,11] เป็นอีกวิธีการหนึ่งซึ่งมีความคล้ายกับ Bagging แต่ AdaBoost จะสร้างตัวแยกแยะที่ต่างกันโดยการจัดการกับชุดข้อมูลสอน (ไม่ใช้การสุ่มข้อมูลใหม่) โดยAdaBoost จะกำหนดน้ำหนัก (weight) ให้กับข้อมูลสอนแต่ละตัวให้มีน้ำหนักแตกต่างกันไป ในแต่ละรอบที่มีการสร้างตัวแยกแยะใหม่ค่าน้ำหนักจะถูกเปลี่ยนไปตามความผิดพลาดของผลลัพธ์ที่ตัวแยกแยะในรอบนั้นๆกระทำต่อข้อมูล ถ้าตัวแยกแยะตอบผลลัพธ์ถูกสำหรับข้อมูลสอนตัวใด ข้อมูลสอนตัวนั้นก็จะถูกลดค่าน้ำหนัก ในทางตรงข้ามถ้าตัวแยกแยะตอบผลลัพธ์ผิดสำหรับข้อมูลสอนตัวใด ข้อมูลสอนนั้นก็จะถูกเพิ่มค่าน้ำหนัก โดยค่าน้ำหนักที่ได้จะใช้สำหรับกำหนดความน่าจะเป็นที่ข้อมูลตัวนั้นจะถูกเลือกให้อยู่ในชุดข้อมูลสอนชุดต่อๆไป ซึ่งในตอนแรกข้อมูลทุกตัวจะถูกกำหนดให้มีค่าความน่าจะเป็นที่จะถูกเลือกเท่ากัน จุดมุ่งหมายของ AdaBoost ก็คือจะให้ความสนใจกับข้อมูลที่ทำนายผิดพลาดมากขึ้นสำหรับการสร้างตัวแยกแยะตัวต่อๆไป

(2) การดำเนินการกับคุณลักษณะของข้อมูลเข้า (Manipulating the Input Features)

วิธีนี้จะทำการดำเนินการกับคุณลักษณะของข้อมูลนำเข้าที่ใช้สำหรับการสอน โดยจะดึงคุณลักษณะประเภทต่างๆจากข้อมูลนำเข้ามาใช้ เช่น อาจใช้ power spectrum based กับ auto-regression based จาก สัญญาณเวลา มาเป็นชุดข้อมูลสำหรับสอน Cherkauer [12] ได้ทำการทดสอบวิธีนี้โดยสร้างระบบ PLANNETT (**P**erson-Level **A**rtificial **N**eural **N**etworks for **E**xtra**T**errestrial **T**errain classification) เพื่อใช้สำหรับการระบุบริเวณที่เป็นภูเขาไฟจากรูปถ่ายพื้นผิวของดาวศุกร์ ซึ่งระบบ PLANNETT ก็สามารถวิเคราะห์บริเวณที่เป็นภูเขาไฟได้ดีเท่าๆกับนักวิจัย โดยทั่วไปวิธีนี้เหมาะกับปัญหาที่มีชุดข้อมูลที่มีคุณสมบัติซ้ำซ้อนกันอยู่เป็นจำนวนมาก

(3) การดำเนินการกับข้อมูลผลลัพธ์ (Manipulate the Output Target)

วิธีนี้จะจัดการกับค่าผลลัพธ์เพื่อใช้ในการสร้างตัวแยกแยะที่แตกต่างกัน Dietterich และ Bakiri [13,14] เสนอวิธีที่เรียกว่า ECOC (**E**rror **C**orrecting **O**utput **C**ode) วิธีนี้เป็นวิธีสำหรับการนำตัวแยกแยะที่แยกเพียง 2 คลาส (two-class) มาใช้สำหรับแก้ปัญหาที่มีมากกว่า 2 คลาส (k-class) ECOC ทำงานโดยการแปลงค่า k คลาสให้เป็นตัวเลขฐาน 2 ที่ไม่ซ้ำกันขนาด n บิต (bit) ตัวเลขฐาน 2 เหล่านี้ถูกเรียกว่า รหัส (codeword) หลังจากนั้นตัวแยกแยะจำนวน n ตัวจะถูกสร้างขึ้นสำหรับแต่ละหลัก ในการหาคำตอบสำหรับปัญหาตัวแยกแยะจำนวน n ตัวจะสร้างข้อมูลเลขฐาน 2 ขนาด n บิต แล้วทำการเปรียบเทียบค่าที่ได้ว่าใกล้เคียงกับรหัสตัวใดมากที่สุดโดยการ

วัดระยะห่างแบบแฮมมิง (Hamming distance) รหัสตัวนั้นจะเป็นคำตอบของปัญหา รูปที่ 2.6 แสดงตัวอย่างของรหัสขนาด 15 บิตสำหรับปัญหาที่มี 10 คลาส

Class	Code Word														
	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}
0	1	1	0	0	0	0	1	0	1	0	0	1	1	0	1
1	0	0	1	1	1	1	0	1	0	1	1	0	0	1	0
2	1	0	0	1	0	0	0	1	1	1	1	0	1	0	1
3	0	0	1	1	0	1	1	1	0	0	0	0	1	0	1
4	1	1	1	0	1	0	1	1	0	0	1	0	0	0	1
5	0	1	0	0	1	1	0	1	1	1	0	0	0	0	1
6	1	0	1	1	1	0	0	0	0	1	0	1	0	0	1
7	0	0	0	1	1	1	1	0	1	0	1	1	0	0	1
8	1	1	0	1	0	1	1	0	0	1	0	0	0	1	1
9	0	1	1	1	0	0	0	0	1	0	1	0	0	1	1

รูปที่ 2.6 รหัสขนาด 15 บิต สำหรับปัญหาที่มี 10 คลาส [14]

(4) วิธีแทรกการสุ่ม (Injecting Randomness)

วิธีแทรกการสุ่มเป็นวิธีทั่วไปที่ใช้ในการสร้างตัวแยกแยะที่ต่างกัน วิธีนี้สามารถใช้ได้กับหลายขั้นตอนวิธีสำหรับการเรียนรู้ (learning algorithm)

สำหรับนิวรอลเน็ตเวิร์กแบบแบคพรอพาเกชัน (Backpropagation Neural Network) Kolen และ Pollack [15] เสนอวิธีพื้นฐานสำหรับการสร้างตัวแยกแยะ โดยแสดงให้เห็นว่าเพียงแค่การกำหนดค่าน้ำหนักเริ่มต้นเปลี่ยนไปก็สามารถสร้างตัวแยกแยะที่ต่างไปจากเดิมได้

นักวิจัยหลายคน [16,17] ได้ทำการเปรียบเทียบวิธีนี้กับ Bagging และ Boosting จากการทดสอบพบว่าวิธีนี้และวิธี Bagging ให้ผลดีเท่าๆกัน

(5) วิธีเฉพาะสำหรับแต่ละปัญหา

บางครั้งวิธีทั่วไปก็ไม่สามารถใช้สร้างตัวแยกแยะที่แตกต่างกันได้จึงมีวิธีที่ถูกพัฒนาขึ้นมาเฉพาะสำหรับปัญหานั้นๆ ซึ่งบางวิธีก็สามารถนำไปใช้กับปัญหาอื่นๆที่ใช้ขั้นตอนวิธีสำหรับการเรียนรู้ (learning algorithm) แบบเดียวกันได้

สำหรับแบคพรอพาเกชันเน็ตเวิร์ก Opitz และ Shavlik [18,19] เสนอวิธีที่เรียกว่า ADDEMUP (Accurate anD Diverse Ensemble-Maker giving United Predictions) วิธีนี้ใช้ขั้นตอนวิธีทางพันธุกรรม (Genetic Algorithm) ในการหาเน็ตเวิร์กที่ถูกต้องและแตกต่างกัน

ADDEMUP ทำงานโดยเริ่มจากการสร้างเน็ตเวิร์กขึ้นมา หลังจากนั้นจะใช้ตัวปฏิบัติการทางพันธุกรรม (Genetic Operators) เช่น การไขว้เปลี่ยน (crossover) หรือ การกลายพันธุ์(mutation) ในการสร้างเน็ตเวิร์กใหม่ขึ้นมา จากนั้น ADDEMUP จะเก็บกลุ่มของเน็ตเวิร์กที่มีความถูกต้องสูงในขณะเดียวกันก็มีข้อผิดพลาดที่ต่างกัน จากการทดสอบพบว่า ADDEMUP เป็นวิธีที่ดีในการสร้างกลุ่มเน็ตเวิร์ก

สำหรับต้นไม้การตัดสินใจ (Decision tree)นั้น มีหลายวิธีที่ถูกคิดขึ้นมาเพื่อแก้ปัญหาเฉพาะ เช่น fBoost โดย Harries [20] วิธีนี้คล้ายกับ Boosting แต่ fBoost ไม่ได้ใช้ค่าความผิดพลาดในการปรับน้ำหนักของแต่ละตัวอย่าง fBoost ใช้ข้อมูลที่ได้จากสมมติฐาน (hypothesis) วิธีนี้แสดงให้เห็นว่าการปรับค่าน้ำหนักไม่จำเป็นที่จะต้องมีการตอบผิด

สำหรับขั้นตอนวิธีอื่นๆ Fan, Stolfo, Zhang, และ Chan [21] เสนอวิธีที่เรียกว่า “AdaCost” วิธีนี้ดัดแปลงมาจาก AdaBoost แต่ AdaCost จะสนใจความยากง่ายของตัวอย่างแต่ละตัว ในแต่ละรอบของการสร้างตัวแยกแยะตัวใหม่ค่าน้ำหนักของตัวอย่างที่ใช้สอนที่ยากต่อการเรียนรู้จะเพิ่มขึ้น และค่าน้ำหนักของตัวอย่างที่ง่ายต่อการเรียนรู้จะลดลง

2.3.2 วิธีรวมผลลัพธ์

มีหลายวิธีในการรวมคำตอบของตัวแยกแยะแต่ละตัว

(1) วิธีไม่ถ่วงน้ำหนัก (Unweighted voting)

วิธีนี้เป็นวิธีที่ง่ายที่สุดในการรวมผลลัพธ์จากตัวแยกแยะแต่ละตัว อย่างไรก็ตามการไม่ใช้น้ำหนักถ่วงนั้นมีข้อดีหลายข้อ [22] วิธีนี้ถูกใช้ในหลายขั้นตอนวิธี (algorithm) เช่น Bagging และ ECOC การรวมแบบไม่ถ่วงน้ำหนักนั้นเหมาะสำหรับตัวแยกแยะที่ให้คำตอบเป็นค่าความน่าจะเป็นที่จะเป็นประเภทนั้นๆ (probability estimate) มากกว่าตัวแยกแยะที่ให้คำตอบเป็นคลาส (simple classification decision) คำตอบของวิธีนี้ได้มาจากคลาส (class) ที่มีความน่าจะเป็นสูงสุด

(2) วิธีถ่วงน้ำหนัก (Weighted voting)

วิธีนี้ดูจะฉลาดกว่าการไม่ถ่วงน้ำหนัก สำหรับการหาคำตอบแบบการตัดสินใจแยกแยะ (classification) ค่าน้ำหนักของตัวแยกแยะแต่ละตัวจะได้มาจากค่าความถูกต้องของตัวแยกแยะบนชุดข้อมูลที่ใช้สอน ยิ่งค่าความถูกต้องสูง ค่าน้ำหนักก็จะยิ่งมาก

Jimenez และ Walsh [23] เสนอการปรับค่าน้ำหนักแบบพลวัต (dynamic) วิธีนี้ค่าน้ำหนักจะเปลี่ยนแปลงขึ้นกับค่าความมั่นใจ (certainty) และน้ำหนักจะถูกคำนวณใหม่ทุกครั้งที่ได้คำตอบ

(3) Gating Network

Gating Network หรือ Gating function วิธีนี้จะใช้ฟังก์ชันในการคำนวณค่าน้ำหนักของตัวแยกแยะแต่ละตัว เช่น soft-max function โดย Jordan และ Jacobs [24]

(4) Stacking

วิธีนี้เป็นวิธีที่ใช้โมเดลระดับสูง (high-level model) ในการรวมผลลัพธ์ของโมเดลระดับต่ำ (low-level model) เพื่อให้ได้ค่าความถูกต้องที่สูงขึ้น [25] สมมุติว่ามีกลุ่มตัวอย่างที่ใช้สอน $\{(x_1,y_1),\ldots,(x_m,y_m)\}$ และมี L ขั้นตอนวิธี A_1, A_2, A_3, ...,A_L จากนั้นจะใช้ ขั้นตอนวิธีเหล่านี้เพื่อสร้างสมมติฐาน $h_1,\ldots,h_L$ จุดมุ่งหมายของวิธีนี้ก็คือการหาตัวแยกแยะ h^* ที่สามารถรวมผลลัพธ์จาก h_1 ถึง h_L ได้ คำตอบของวิธีนี้หาได้จาก $h^*(h_1(x),\ldots,h_L(x))$

2.3.3 ข้อดีของการใช้กลุ่มก้อนของตัวแยกแยะ

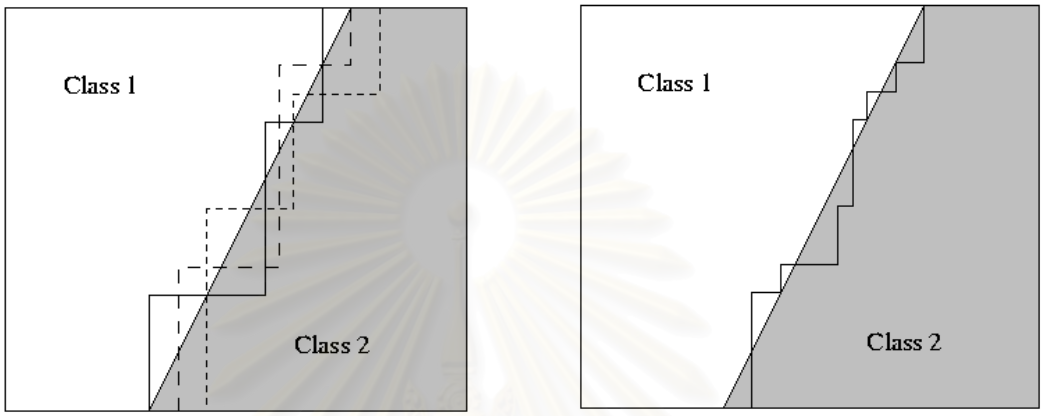
Dietterich [26] เสนอให้เห็นถึงสาเหตุสำคัญอย่างน้อย 3 ประการที่ทำให้การใช้กลุ่มก้อนของตัวแยกแยะเป็นทางเลือกที่ดี ที่จะใช้ในการปรับปรุงประสิทธิภาพของการตัดสินแยกแยะ

(1) ชุดข้อมูลที่ใช้สอนอาจไม่มีข้อมูลเพียงพอสำหรับการเลือกสมมติฐานที่ดีที่สุดจากกลุ่มของสมมติฐาน (hypothesis space) อย่างไรก็ตามการเลือกสมมติฐานที่ดีที่สุดจากกลุ่มของสมมติฐานนี้ไม่ใช่วิธีที่ดีที่สุด เพราะการไม่สนใจสมมติฐานที่ด้อยกว่าอาจเป็นการทิ้งข้อมูลที่มีค่าอื่นๆ จากจุดนี้วิธีการใช้กลุ่มก้อนของตัวแยกแยะก็จะเป็นทางออกที่ดีสำหรับปัญหานี้

(2) ขั้นตอนวิธีการเรียนรู้อาจไม่สามารถหาสมมติฐานที่ดีที่สุดได้เนื่องจากปัญหาเรื่องความยากของการค้นหา (search problem) ดังนั้นการสร้างสมมติฐานที่หลากหลายและรวมเข้าด้วยกันอาจสร้างสมมติฐานที่ซับซ้อนขึ้นได้ ซึ่งการใช้กลุ่มก้อนของตัวแยกแยะสามารถแก้ปัญหานี้ได้

(3) ในกลุ่มของสมมติฐานอาจไม่มีสมมติฐานที่แท้จริง แต่มีเพียงสมมติฐานที่ใกล้เคียงอยู่ โดยการสร้างสมมติฐานหลายๆสมมติฐานอาจสามารถสร้างสมมติฐานที่ใกล้เคียงกับสมมติฐาน

จริงได้ ซึ่งแสดงให้เห็นได้ดังรูปที่ 2.7 รูปซ้ายแสดงเส้นแบ่งขอบเขต (Decision boundary) ที่แท้จริงของข้อมูล 2 ประเภท และเส้นขั้นบันได 3 เส้นที่ประมาณค่าของเส้นแบ่งขอบเขต ส่วนรูปขวาแสดงเส้นแบ่งขอบเขตที่ได้จากการรวมกลุ่มก้อนของตัวแยกแยะ ซึ่งใกล้เคียงกับเส้นแบ่งขอบเขตที่แท้จริงมากกว่า



รูปที่ 2.7 แสดงให้เห็นเส้นแบ่งขอบเขตที่ได้จากการรวมกลุ่มก้อนของตัวแยกแยะว่ามีความใกล้เคียงกับสมมติฐานจริงมากกว่าสมมติฐานจากตัวแยกแยะเพียงตัวเดียว

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 3

การออกแบบและพัฒนา

ในงานวิจัยนี้เราได้ใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์กเพื่อหาวิธีที่เหมาะสมในการรู้จำตัวพิมพ์อักษรไทย โดยเราได้ทดสอบกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างและรวมผลลัพธ์เข้าด้วยกันในหลายๆวิธี

3.1 ข้อกำหนดต่าง ๆของนิวรอลเน็ตเวิร์ก

ในวิทยานิพนธ์ฉบับนี้เราใช้นิวรอลเน็ตเวิร์กที่มีโครงสร้างเหมือนกับงานวิจัยของธเนศ[2]คือ กำหนดให้

1. จำนวนโนดในชั้นอินพุตเท่ากับ 128 โนด เนื่องจากใช้เวกเตอร์รูปแบบที่มีจำนวนสมาชิกเป็น 128 ตัว
2. จำนวนโนดในชั้นซ่อนเท่ากับ 128 โนด เป็นค่าที่ได้จากการทดลอง
3. จำนวนโนดในชั้นเอาต์พุตเท่ากับ 68 โนด ซึ่งเท่ากับจำนวนตัวอักษรที่สอนให้นิวรอลเน็ตเวิร์กรู้จำ

การทดลองได้ใช้เงื่อนไขในการเรียนรู้เหมือนกันทุกเน็ตเวิร์กดังนี้

1. ค่าอัตราการเรียนรู้ (Learning Rate) เท่ากับ 0.08
2. ค่าอินเนอร์เชียหรือโมเมนตัม (Inertia or Momentum) เท่ากับ 0.95
3. ใช้ฟังก์ชันกระตุ้น (Activation Function) แบบซิกมอยด์ฟังก์ชัน (Sigmoid Function) ซึ่งให้ค่าของฟังก์ชันอยู่ในช่วง (0,1)
4. เงื่อนไขในการลู่เข้า (convergence) คือ ต้องเรียนรู้ตัวอย่างให้ถูกต้องเป็นจำนวน
   - 3000 ตัวอย่าง โดยมีค่าผิดพลาดไม่เกิน 0.5 สำหรับการสร้างเน็ตเวิร์กด้วยวิธี Bagging
   - 500 ตัวอย่าง โดยมีค่าผิดพลาดไม่เกิน 0.5 สำหรับการสร้างเน็ตเวิร์กด้วยวิธีแบ่งตามรูปแบบของตัวอักษร
5. ค่าเริ่มต้นของค่าน้ำหนักและไบแอสได้จากการสุ่มค่าระหว่าง [-0.1,0.1]

การพิจารณาผลลัพธ์ของเน็ตเวิร์กนั้น จะกำหนดให้โนดที่มีค่าเอาต์พุตสูงที่สุดเพียงโนดเดียวเป็นคำตอบ เช่น กำหนดว่าคำตอบของการรู้จำ รูปตัวอักษร ก นั้น โนดที่ 1 จะต้องมีค่าสูงที่สุด ซึ่งถ้าเน็ตเวิร์กใดมี โนดอื่นที่มีค่าสูงกว่าค่าของโนดที่ 1 แล้วจะถือว่าเน็ตเวิร์กนั้นตอบผิด

3.2 วิธีที่ใช้ในการสร้างกลุ่มก้อนของนิวรอลเน็ตเวิร์ก

วิธีที่เราใช้ในการสร้างกลุ่มก้อนของตัวแยกแยะมี 2 วิธี คือ (1) แบ่งตามรูปแบบของตัวอักษร (font) (2) Bagging

(1) วิธีแบ่งตามรูปแบบของตัวอักษร ทำได้โดยการแบ่งชุดข้อมูลสอนออกเป็น 6 ชุด ตามรูปแบบของตัวอักษร คือ Angsana Browalia Cordia Dillenia Eucrosia และ Freesia ดังนั้นชุดข้อมูลสำหรับสอนสำหรับนิวรอลเน็ตเวิร์กแต่ละเน็ตเวิร์กจะประกอบด้วย ตัวอักษร 68 ตัว แบ่งเป็น พยัญชนะ 44 ตัว สระและวรรณยุกต์รวม 24 ตัว โดยแต่ละตัวมี 8 ขนาด รวมทั้งหมดเป็น 544 ตัว

(2) วิธี Bagging ทำได้โดยการสร้างชุดข้อมูลสอนใหม่จากการสุ่มข้อมูลในชุดข้อมูลสอนเดิม แล้วนำชุดข้อมูลสอนนั้นมาทำการสอนนิวรอลเน็ตเวิร์ก ซึ่งวิธีนี้เป็นวิธีสร้างกลุ่มก้อนของตัวแยกแยะที่เหมาะสำหรับนิวรอลเน็ตเวิร์ก ชุดข้อมูลสอนสำหรับนิวรอลเน็ตเวิร์กแต่ละเน็ตเวิร์กจะประกอบด้วย ตัวอักษร 3264 ตัว

สาเหตุที่ไม่เลือกใช้วิธี AdaBoost ถึงแม้ว่าจากผลการวิจัยที่ผ่านมา [10,16] ได้แสดงให้เห็นถึงความมีประสิทธิภาพของวิธีนี้นั้น เนื่องจากวิธี AdaBoost ไม่เหมาะสำหรับการรู้จำข้อมูลชุดนี้เพราะว่า ในงานวิจัยนี้นิวรอลเน็ตเวิร์กเดี่ยวมีประสิทธิภาพสูงมาก ทำให้เกิดข้อผิดพลาดของข้อมูลสอนน้อยมากหรือไม่มีเลย ทำให้ AdaBoost ไม่สามารถทำงานได้เนื่องจากการปรับค่าน้ำหนักของวิธี AdaBoost นั้นขึ้นอยู่กับค่าความผิดพลาดของตัวแยกแยะในขั้นตอนการสอน ดังนั้นนิวรอลเน็ตเวิร์กที่ได้จากวิธี AdaBoost นี้จึงไม่มีความแตกต่างจากวิธี Bagging เลย นอกจากนี้วิธีดำเนินการกับคุณลักษณะของข้อมูลเข้าก็ไม่เหมาะที่จะนำมาใช้ เนื่องจากข้อมูลชุดนี้ไม่ได้มีคุณสมบัติที่ซ้ำซ้อนกันเป็นจำนวนมาก ส่วนวิธีดำเนินการกับข้อมูลผลลัพธ์นั้น เนื่องข้อมูลที่ใช้นั้นมีจำนวนคลาสถึง 68 คลาส หากต้องการใช้วิธี ECOC เราจะต้องใช้นิวรอลเน็ตเวิร์กเป็นจำนวนมาก ดังนั้นเราจึงเลือกใช้วิธี Bagging แล้วนำมาปรับเพื่อให้เหมาะกับชุดข้อมูลของตัวพิมพ์อักษรไทยที่เราต้องการ

3.3 วิธีที่ใช้ในการรวมผลลัพธ์จากกลุ่มก้อนของนิวรอลเน็ตเวิร์ก

เมื่อเราสร้างกลุ่มก้อนของนิวรอลเน็ตเวิร์กได้แล้ว ขั้นตอนต่อมาก็คือการรวมคำตอบของนิวรอลเน็ตเวิร์กแต่ละตัว วิธีที่เราใช้ในการรวมผลลัพธ์ของตัวแยกแยะมี 6 วิธี ซึ่งสามารถแบ่งได้เป็น 3 ประเภท คือ

1) วิธีไม่ถ่วงน้ำหนัก (Unweighted Voting)

วิธีนี้ได้แก่วิธี (1) Majority Vote: MV (2) Majority Vote with Probability: MVP (3) Average Voting: AV

2) วิธีถ่วงน้ำหนัก (Weighted voting)

วิธีนี้ได้แก่วิธี (4) Minimum Margin Weighted Voting: MMWV และ (5) Class-based Weighted Voting: CWV

3) วิธีเฉพาะปัญหา (Specific)

วิธีนี้ได้แก่วิธี (6) Hierarchy Voting: HV

ในจำนวนวิธีทั้ง 6 วิธีนี้ MMWV, CWV, และ HV เป็นวิธีการที่นำเสนอโดยวิทยานิพนธ์ฉบับนี้ ส่วนวิธีการอื่นๆเป็นวิธีดั้งเดิมที่มีอยู่แล้ว ซึ่งแต่ละวิธีมีรายละเอียดดังนี้

(1) Majority Vote (MV)

วิธีนี้จะคิดเป็นคะแนนเสียงของนิวรอลเน็ตเวิร์ก แต่ละเน็ตเวิร์กมี 1 เสียง โดยเน็ตเวิร์กแต่ละตัวจะโหวตให้กับคลาสที่มีค่าเอาต์พุตสูงสุดเป็น 1 คะแนน และโหวตให้กับคลาสอื่นๆเป็น 0 คะแนน คลาสไหนที่มีคะแนนโหวตมากที่สุดถือเป็นคำตอบ

(2) Majority Vote with Probability (MVP)

วิธีนี้จะนำเอาค่าเอาต์พุตที่มากที่สุดที่ได้จากนิวรอลเน็ตเวิร์กทุกเน็ตเวิร์กมารวมกัน คลาสไหนที่มีค่าเอาต์พุตโดยรวมมากที่สุดถือเป็นคำตอบ

(3) Average Voting (AV)

วิธีนี้จะนำเอาค่าเอาต์พุตที่ได้จากโนดทุกโนดของนิวรอลเน็ตเวิร์กทุกเน็ตเวิร์กมาหาค่าเฉลี่ย คลาสไหนที่มีค่าเอาต์พุตเฉลี่ยมากที่สุดถือเป็นคำตอบ

(4) Mimimum Margin Weighted Voting (MMWV)

วิธีนี้จะกำหนดค่าน้ำหนักสำหรับนิวรอลเน็ตเวิร์กแต่ละเน็ตเวิร์ก เปลี่ยนไปตามค่าความแตกต่างระหว่างค่าเอาต์พุตของนิวรอลเน็ตเวิร์กที่มีค่ามากที่สุดกับค่าที่มีค่ามากรองลงมาเป็นอันดับที่สอง เพราะนิวรอลเน็ตเวิร์กที่น่าเชื่อถือควรจะมีค่าความแตกต่างระหว่างค่าเอาต์พุตของคำตอบที่มากที่สุดกับค่าเอาต์พุตที่มากรองลงมาสูง คำตอบของวิธีนี้สามารถหาได้จากการนำค่าเอาต์พุตของนิวรอลเน็ตเวิร์กที่มีค่ามากที่สุดมาคูณกับค่าน้ำหนักซึ่งเท่ากับค่าความแตกต่างระหว่างค่าเอาต์พุตของนิวรอลเน็ตเวิร์กที่มีค่ามากที่สุดกับค่าที่มีค่ามากรองลงมา คลาสใดที่มีค่าเอาต์พุตโดยรวมมากที่สุดถือเป็นคำตอบ

(5) Class-based Weighted Voting (CWV) : **วิธีโหวตแบบถ่วงน้ำหนักโดยใช้คลาส**

วิธี CWV นี้มีประโยชน์มากในกรณีที่ต้องการรวมผลลัพธ์ด้วยวิธีถ่วงน้ำหนักแต่ไม่สามารถหาค่าน้ำหนักของตัวแยกแยะแต่ละตัวได้เนื่องจาก ตัวแยกแยะที่มีอยู่มีความถูกต้องสูงมากจนทำให้ไม่มีความผิดพลาดเลย ดังนั้นค่าน้ำหนักสำหรับตัวแยกแยะแต่ละตัวจึงเท่ากันหมด แต่ถ้าหากใช้วิธี CWV นี้ แม้ว่าตัวแยกแยะจะไม่มีค่าความผิดพลาดเลยก็ยังสามารถที่จะกำหนดน้ำหนักให้กับตัวแยกแยะแต่ละตัวได้

นอกจากนี้วิธี CWV ยังไม่เพียงกำหนดน้ำหนักให้กับตัวแยกแยะแต่ละตัวเท่านั้น แต่น้ำหนักของตัวแยกแยะแต่ละตัวจะถูกเปลี่ยนไปโดยขึ้นอยู่กับคำตอบที่ได้อีกด้วย ซึ่งจะแตกต่างจากการกำหนดค่าน้ำหนักของตัวแยกแยะในแบบเดิมที่ใช้เพียงค่าความผิดพลาดของตัวแยกแยะบนชุดข้อมูลสอนจึงเป็นการละทิ้งข้อมูลที่สำคัญคือความถนัดของตัวแยกแยะสำหรับปัญหาแต่ละปัญหา นั่นคือเป็นการคิดเสมือนว่าตัวแยกแยะแต่ละตัวได้เรียนรู้คลาสทุกๆคลาสอย่างเท่าเทียมกัน แต่สำหรับการสร้างตัวแยกแยะโดยการใช้วิธี Bagging นั้น จำนวนครั้งที่แต่ละคลาสจะถูกเลือกมาสอนตัวแยกแยะแต่ละตัวนั้นไม่จำเป็นจะต้องเท่ากันเนื่องจากใช้การสุ่มข้อมูลสอน ดังนั้นการกำหนดให้ค่าน้ำหนักของตัวแยกแยะแต่ละตัวขึ้นอยู่กับคำตอบของตัวแยกแยะที่ได้จะเป็นการกำหนดค่าน้ำหนักได้ดีกว่าการใช้ค่าความผิดพลาดรวมของทุกๆคลาสมาเป็นตัวกำหนดค่าน้ำหนัก

วิธีการที่นำเสนอนี้จะปรับค่าน้ำหนักของนิวรอลเน็ตเวิร์กไปตามสัดส่วนของจำนวนครั้งที่เน็ตเวิร์กนั้นได้ทำการเรียนรู้คลาส (class) ของข้อมูล ในกรณีของการรู้จำตัวอักษรไทยที่ทำการทดลองในบทที่ 4 นั้นจะมีอยู่ทั้งหมด 68 คลาส คือ ‘ก’, ‘ข’,… ,‘ฮ’, ‘ะ’,… , ‘ˆ’,

ยิ่งเน็ตเวิร์กได้เรียนคลาสนั้นบ่อยเท่าใด ค่าน้ำหนักก็จะยิ่งมากขึ้นเท่านั้น เนื่องจากเรามีสมมติฐานที่ว่า หากเน็ตเวิร์กถูกสอนด้วยคลาสใดมากๆ เมื่อเน็ตเวิร์กตอบคลาสนั้นก็ย่อมจะมีความน่าเชื่อถือกว่าคำตอบเดียวกันที่ได้จากเน็ตเวิร์กที่ถูกสอนน้อยครั้งกว่า เราเรียกวิธีการที่นำเสนอนี้ว่า วิธีโหวตแบบถ่วงน้ำหนักโดยใช้คลาส ขั้นตอนวิธีของวิธีนี้ แสดงให้เห็นในรูปที่ 3.1

Algorithm: CWV

m : จำนวนคำตอบ, t : จำนวนเน็ตเวิร์ก, s : จำนวนคลาส

$posterior_{kl}$: ค่าเอาต์พุตของคลาส k ที่ได้จากเน็ตเวิร์ก l

1. ให้ C เป็นเซ็ตของคลาสทั้งหมดที่ทำการสอนให้กับเน็ตเวิร์ก

$$C = \{c_1, c_2, c_3, ..., c_s\}$$

2. ให้ N เป็นเซ็ตของเน็ตเวิร์กทั้งหมด

$$N = \{n_1, n_2, n_3, ..., n_t\}$$

3. ให้ A เป็นเซ็ตของคำตอบที่ได้จากเน็ตเวิร์กทั้งหมด เพื่อดูว่ามีคำตอบที่แตกต่างกันทั้งหมดอยู่ในคลาสใดบ้าง โดยที่ $A \subseteq C$

4. ให้ M เป็นเซ็ตของจำนวนครั้งที่แต่ละคลาสถูกสอนให้กับแต่ละเน็ตเวิร์ก

$$M = \{m_{c_1 n_1}, m_{c_2 n_1}, ..., m_{c_s n_1}, m_{c_1 n_2}, m_{c_2 n_2}, ..., m_{c_s n_t}\}$$

โดยที่ $m_{c_i n_j}$ คือ จำนวนครั้งที่เน็ตเวิร์ก n_j ถูกสอนด้วยคลาส c_i

5. คำนวณค่าน้ำหนักของแต่ละคลาสของแต่ละเน็ตเวิร์ก ($W_{c_i n_j}$)จากสูตร

$$W_{c_i n_j} = \frac{m_{c_i n_j}}{\sum_{j=1}^{n} m_{c_i n_j}}$$

$W_{c_i n_j}$ คือ อัตราส่วนของจำนวนครั้งที่เน็ตเวิร์ก n_j ถูกสอนด้วยคลาส c_i ต่อ จำนวนครั้งที่ทุกเน็ตเวิร์กถูกสอนด้วยคลาส c_i รวมกัน

6. คำนวณค่าผลลัพธ์ของแต่ละคลาส (P_k) จากสูตร

$$P_k = \sum_{l=1}^{n} \left(posterior_{kl} \times W_{kl} \right)$$

โดยที่ $k \in A$ และ $l \in N$

7. คำตอบที่ได้คือคลาสที่มีค่าผลลัพธ์ (P_k) สูงสุด

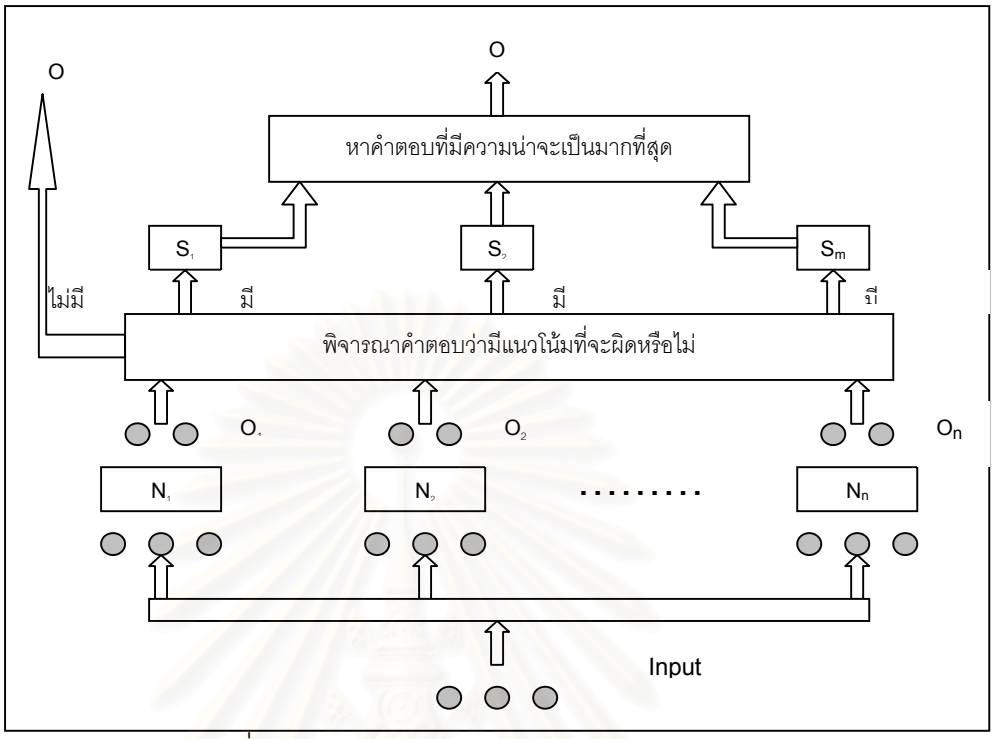
รูปที่ 3.1 ขั้นตอนวิธีของวิธีโหวตแบบถ่วงน้ำหนักโดยใช้คลาส

(6) Hierarchy Voting (HV)

วิธี HV นี้มีประโยชน์ในกรณีที่การสร้างกลุ่มก้อนไม่สามารถสร้างตัวแยกแยะที่มีความผิดพลาดที่ต่างกันได้ จึงทำให้ตัวแยกแยะแต่ละตัวยังมีความผิดพลาดที่ซ้ำซ้อนกันอยู่ ทำให้ถึงแม้จะใช้กลุ่มก้อนก็ไม่สามารถจะแก้ปัญหาในกรณีนี้ได้ ดังนั้นการสร้างตัวแยกแยะที่มีความถนัดสำหรับแก้ปัญหาในกรณีที่เกิดการผิดพลาดซ้ำซ้อนกันนี้โดยเฉพาะจึงสามารถช่วยลดความผิดพลาดได้

วิธี HV นี้จะมีการทำงานโดยใช้กลุ่มก้อนของเน็ตเวิร์ก 2 กลุ่ม กลุ่มแรกจะเป็นเน็ตเวิร์กที่สร้างจากวิธี Bagging โดยใช้ชุดข้อมูลในการสอนคือ ตัวอักษร 68 ตัวจาก 6 รูปแบบ (font) 8 ขนาด (size) รวมเป็น 3264 ตัวอย่าง กลุ่มที่สองจะเป็นเน็ตเวิร์กเฉพาะทาง (Specific network) ซึ่งเน็ตเวิร์กแต่ละตัวจะถูกสอนโดยใช้ชุดข้อมูลในการสอนต่างๆกัน การเลือกข้อมูลสำหรับสอนนั้นสามารถทำได้ 2 วิธี คือ (1) การเลือกด้วยมือ และ (2) การเลือกอัตโนมัติโดยโปรแกรม สำหรับการเลือกด้วยมือนั้น ชุดข้อมูลจะถูกเลือกให้ตัวที่มักจะถูกตอบผิดกับตัวคำตอบที่ผิด หรือเลือกตัวอักษรที่มีลักษณะคล้ายกัน อยู่ในชุดข้อมูลสำหรับใช้สอนเน็ตเวิร์กเดียวกัน เช่นจัด ตัว ฎ, ฏ, ฤ และ ฦ อยู่ในชุดเดียวกัน สำหรับการเลือกอัตโนมัติโดยโปรแกรมนั้น โปรแกรมจะทำการเลือกข้อมูลที่มีค่าใกล้เคียงกันอยู่ในชุดข้อมูลสำหรับสอนเดียวกัน เพื่อที่เน็ตเวิร์กเฉพาะทางจะได้แยกความแตกต่างของตัวอักษรเหล่านี้ได้ วิธีการทำงานของวิธี Hierachy สามารถดูได้ที่รูปที่ 3.2

ในการหาคำตอบของวิธีนี้นั้นจะมีการตัดสินแยกแยะคำตอบ 2 ขั้น ในขั้นแรกจะใช้กลุ่มของเน็ตเวิร์ก n เน็ตเวิร์ก (N_1, N_2, N_3,…,N_n) ที่สร้างด้วยวิธีใดๆ เพื่อหาคำตอบ หลังจากนั้นค่าคำตอบที่ได้ออกมาจากเน็ตเวิร์กแต่ละตัว (O_1, O_2, O_3,…, O_n) จะถูกนำไปพิจารณาว่าค่าคำตอบนั้นมีแนวโน้มที่จะผิดหรือไม่ ถ้าไม่มีก็สามารถตอบได้เลย ไม่ต้องนำไปผ่านเน็ตเวิร์กเฉพาะทาง (S_1, S_2, S_3,…,S_m) อีก แต่ถ้ามีจะตรวจสอบดูว่าค่าคำตอบที่ได้นั้นอยู่ในความรับผิดชอบของเน็ตเวิร์กเฉพาะทางตัวใด จากนั้นจะส่งข้อมูลเข้าไปให้เน็ตเวิร์กเฉพาะทางนั้นๆจำแนกประเภท ถ้าคำตอบที่ได้จากเน็ตเวิร์กเฉพาะทางตัวใดสูงสุดก็ใช้ค่าคำตอบจากเน็ตเวิร์กตัวนั้นเป็นคำตอบของกลุ่มก้อน ถ้าการแบ่งกลุ่มของชุดข้อมูลในการสร้างเน็ตเวิร์กเฉพาะทางเหมาะสม ค่าคำตอบจากเน็ตเวิร์กในขั้นแรกควรจะอยู่ในเน็ตเวิร์กเฉพาะทางเพียงเน็ตเวิร์กเดียว ซึ่งจะทำให้วิธีนี้มีประสิทธิภาพสูงสุด



รูปที่ 3.2 แสดงวิธีการทำงานของวิธี Hierarchy Voting

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 4

การทดลองและผลการทดลอง

4.1 แหล่งที่มาของข้อมูลภาพ

ข้อมูลที่ใช้ทดสอบการรู้จำตัวพิมพ์อักษรไทยเป็นข้อมูลภาพที่เก็บไว้ในแฟ้มข้อมูล 1 แฟ้มต่อหนึ่งตัวอักษร การจัดเก็บข้อมูลภาพของตัวอักษรจัดเก็บโดยใช้วิธีการจัดเก็บแบบ Windows Bitmap (BMP) โดยตัวอักษรที่ใช้ในการวิจัยนี้ใช้รูปแบบของทรูไทป์ที่มีอยู่บนระบบปฏิบัติการวินโดวส์ โดยการทดลองนี้ใช้ตัวอักษร 6 รูปแบบคือ AngsanaUPC, BrowaliaUPC, CordiaUPC, DilleniaUPC, EucrosiaUPC และ FreesiaUPC โดยรูปแบบแต่ละรูปแบบจะมีตัวอักษรทั้งหมด 68 ตัวแบ่งออกเป็น 2 กลุ่มใหญ่ๆ ได้ดังนี้

- พยัญชนะ 44 ตัวอักษร ได้แก่
  ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
- สระและวรรณยุกต์รวม 24 ตัวอักษร ได้แก่
  ะ า ิ ี ึ ื ุ ู ็ ั ์ เ แ ไ ใ ่ ้ ๊ ๋ ๆ ฯ ่ ้ ๊ ๋ ็

นอกจากนี้แต่ละรูปแบบของตัวอักษรจะประกอบด้วยตัวอักษรขนาด 16, 18, 20, 22, 24, 26, 28 และ 36 จุด ดังนั้นในหนึ่งชุดตัวอักษรจะมีทั้งหมด 3264 ตัวอักษร ตัวอย่างตัวอักษรที่ใช้สามารถดูได้ที่ภาคผนวก ก

ข้อมูลมีทั้งหมด 3 ชุดดังนี้

1. *ชุดสอน*: ชุดข้อมูลต้นแบบสำหรับใช้สอน ได้มาจากการพิมพ์ตัวอักษรทั้งหมด 3264 ตัวอักษรจากโปรแกรม Microsoft Word for Windows 95 version 7.0 ด้วยเครื่องพิมพ์แบบเลเซอร์ที่ความละเอียด 600 จุดต่อนิ้ว แล้วนำเอกสารที่ได้ไปทำการอ่านด้วยเครื่องสแกนเนอร์โดยใช้ความละเอียด 200 จุดต่อนิ้ว แล้วทำการตัดตัวอักษรเก็บลงแฟ้มข้อมูล 1 แฟ้มต่อหนึ่งตัวอักษร

2. *ชุดทดสอบที่ 1*: ข้อมูลสำหรับใช้ทดสอบชุดที่ 1 ได้มาจากการนำเอกสารที่ได้มาจากชุดสอนมาทำการถ่ายเอกสารด้วยเครื่องถ่ายเอกสารโดยปรับความเข้มของการถ่ายให้จางลง แล้วนำเอกสารที่ได้จากการถ่ายเอกสารไปทำการอ่านด้วยเครื่องสแกนเนอร์ที่ความละเอียด 200 จุดต่อนิ้ว แล้วทำการตัดตัวอักษรเก็บลงแฟ้มข้อมูล 1 แฟ้มต่อหนึ่งตัวอักษร

3. *ชุดทดสอบที่ 2*: ข้อมูลสำหรับใช้ทดสอบชุดที่ 2 ได้มาจากการนำเอกสารที่ได้มาจากชุดสอนมาทำการถ่ายเอกสารด้วยเครื่องถ่ายเอกสารโดยปรับความเข้มของการถ่ายให้เข้มขึ้น แล้วนำเอกสารที่ได้จากการถ่ายเอกสารไปทำการอ่านด้วยเครื่องสแกนเนอร์ที่ความละเอียด 200 จุดต่อนิ้ว แล้วทำการตัดตัวอักษรเก็บลงแฟ้มข้อมูล 1 แฟ้มต่อหนึ่งตัวอักษร

4.2 วิธีการทดลอง

4.2.1 <u>การสร้างกลุ่มก้อนของนิวรอลเน็ตเวิร์ก</u>

1. สร้างโดยการแบ่งชุดข้อมูลสอนออกเป็น 6 ชุด แยกตามรูปแบบของตัวอักษร แต่ละชุดจะประกอบด้วยตัวอักษรทั้งหมด 544 ตัว คือ ตัวอักษร 68 ตัว ตัวละ 8 ขนาด แล้วทำการสร้างนิวรอลเน็ตเวิร์กสำหรับทุกชุด รวมเป็น 6 เน็ตเวิร์ก โดยกำหนดค่าตัวแปรต่างๆ สำหรับนิวรอลเน็ตเวิร์กตามรายละเอียดในบทที่ 3
2. สร้างด้วยวิธี Bagging โดยทำการสร้างนิวรอลเน็ตเวิร์กทั้งหมด 21 เน็ตเวิร์ก โดยกำหนดค่าตัวแปรต่างๆสำหรับนิวรอลเน็ตเวิร์กตามรายละเอียดในบทที่ 3

4.2.2 <u>การรวมผลลัพธ์จากกลุ่มก้อนของนิวรอลเน็ตเวิร์ก</u>

จะรวมด้วยวิธีที่เสนอไปในบทที่ 3 ดังต่อไปนี้

(1) Majority Vote: MV
(2) Majority Vote with Probability: MVP
(3) Average Voting: AV
(4) Minimum Margin Weighted Voting: MMWV
(5) Class-based Weighted Voting: CWV
(6) Hierarchy Voting: HV

สำหรับการรวมแบบ HV นั้น เราได้แบ่งกลุ่มของเน็ตเวิร์กเฉพาะทางดังนี้

1. การเลือกด้วยมือ เป็นการแบ่งโดยใช้ความคล้ายกันของตัวอักษรเป็นหลักในการแบ่ง โดยจะจัดให้ตัวอักษรที่มักจะถูกตอบผิดและคำตอบที่ผิดอยู่ในกลุ่มเดียวกัน เช่น ค มักจะถูกตอบผิดเป็น ก ดังนั้นจึงจัดให้ ค และ ก อยู่ในกลุ่มเดียวกัน จัดแบ่งได้เป็น 12 กลุ่ม โดยแต่ละกลุ่มมีสมาชิกดังนี้

กลุ่มที่ 1: ก ค ศ ฎ ฏ ฒ ด ต ถ ภ ฤ ฦ ศ จำนวน 13 ตัว
กลุ่มที่ 2: ข ฆ ช ซ จำนวน 4 ตัว
กลุ่มที่ 3: ง จ ฐ ธ ร จำนวน 5 ตัว
กลุ่มที่ 4: ฆ ฉ น ม จำนวน 4 ตัว
กลุ่มที่ 5: ฌ ญ ณ จำนวน 3 ตัว
กลุ่มที่ 6: ฑ ท ห จำนวน 3 ตัว
กลุ่มที่ 7: บ ป ผ ฝ พ ฟ ษ ย ฬ จำนวน 9 ตัว
กลุ่มที่ 8: ล ส จำนวน 2 ตัว
กลุ่มที่ 9: ว อ ฮ จำนวน 3 ตัว
กลุ่มที่ 10: ฯ ะ า เ โ ไ ใ ๆ จำนวน 8 ตัว
กลุ่มที่ 11: [illegible] จำนวน 12 ตัว
กลุ่มที่ 12: ุ ู จำนวน 2 ตัว

2. การเลือกอัตโนมัติโดยโปรแกรม

ในงานวิจัยนี้เราใช้โปรแกรม Autoclass version 3.3.3[1] ซึ่งเรากำหนดให้อินพุตของโปรแกรมคือค่าเวกเตอร์รูปแบบที่มีสมาชิก 128 ตัว เมื่อสั่งให้โปรแกรม Autoclass ทำงาน โปรแกรมทำการแบ่ง ข้อมูลออกได้เป็น 9 กลุ่ม โดยแต่ละกลุ่มมีสมาชิกดังนี้

กลุ่มที่ 1: ก ง ฉ ญ ฎ ฏ ฐ ป ฝ พ ฟ ร ฤ ภ ว ฬ ะ า [illegible] จำนวน 32 ตัว
กลุ่มที่ 2: ท น บ ภ ม ษ ห จำนวน 7 ตัว
กลุ่มที่ 3: ข ฆ ฒ ช ซ ฑ + จำนวน 7 ตัว
กลุ่มที่ 4: จ ธ ย อ ฮ จำนวน 5 ตัว
กลุ่มที่ 5: ถ ล ศ ส จำนวน 4 ตัว
กลุ่มที่ 6: [illegible] จำนวน 4 ตัว
กลุ่มที่ 7: ค ผ ฯ จำนวน 3 ตัว
กลุ่มที่ 8: ฌ ฒ ณ จำนวน 3 ตัว
กลุ่มที่ 9: ศ ด ต จำนวน 3 ตัว

[1] สามารถดาวน์โหลดได้จาก http://ic-www.arc.nasa.gov/ic/projects/bayes-group/autoclass/autoclass-c-program.html

การที่จะให้เน็ตเวิร์กเฉพาะทางสำหรับวิธี HV สามารถทำงานได้อย่างมีประสิทธิภาพจะต้องมีการกำหนดค่าต่างๆ 3 ค่า ซึ่งค่าเหล่านี้จะต้องกำหนดให้เหมาะสม คือ (1) ค่าขอบเขตบน (2) ค่าขอบเขตล่าง และ (3) ค่าขีดแบ่ง (threshold) โดยค่าแต่ละค่ามีนิยามดังนี้

(1) ค่าขอบเขตบน คือ ค่าเอาต์พุตที่มากที่สุดที่กำหนดให้เน็ตเวิร์กเฉพาะทางทำงาน หากค่าเอาต์พุตที่สูงที่สุดมีค่ามากกว่าค่าขอบเขตบนนี้ เน็ตเวิร์กเฉพาะทางจะไม่ทำงาน เพราะหากค่าเอาต์พุตที่มากที่สุดมีค่ามากกว่าค่าขอบเขตบนที่กำหนดจะถือว่าค่าคำตอบนั้นน่าจะถูกแล้ว

(2) ค่าขอบเขตล่าง คือ ค่าเอาต์พุตที่น้อยที่สุดที่กำหนดให้เน็ตเวิร์กเฉพาะทางทำงาน หากค่าเอาต์พุตที่สูงที่สุดมีค่าน้อยกว่าค่าขอบเขตล่างนี้ เน็ตเวิร์กเฉพาะทางจะไม่ทำงาน เพราะหากค่าเอาต์พุตที่มากที่สุดมีค่าน้อยกว่าค่าขอบเขตล่างที่กำหนด การรวมโดยวิธี AV จะให้ผลที่ดีกว่าการให้เน็ตเวิร์กเฉพาะทางทำงาน

(3) ค่าขีดแบ่ง คือ ค่าที่กำหนดให้เน็ตเวิร์กเฉพาะทางทำงานถ้าค่าความแตกต่างระหว่างค่าเอาต์พุตที่มากที่สุดและเอาต์พุตที่มากรองลงมาเป็นอันดับที่สองมีค่าน้อยกว่าค่าขีดแบ่งที่กำหนด ซึ่งค่าความแตกต่างนี้แสดงให้เห็นถึงความน่าจะเป็นที่คำตอบจะผิด หากค่าความแตกต่างมากแสดงว่าคำตอบน่าจะถูก แต่หากค่าความแตกต่างน้อยแสดงว่าคำตอบน่าจะผิด

ดังนั้นเน็ตเวิร์กเฉพาะทางจะทำงานก็ต่อเมื่อ ค่าเอาต์พุตที่มากที่สุดหลังจากการรวมผลลัพธ์โดยวิธี AV มีค่าอยู่ในช่วงของค่าขอบเขตบนและค่าขอบเขตล่าง และมีค่าความแตกต่างระหว่างค่าเอาต์พุตที่มากที่สุดและเอาต์พุตที่มากรองลงมาน้อยกว่าค่าขีดแบ่งที่กำหนด

4.3 ผลการทดลอง

4.3.1 ทดสอบนิวรอลเน็ตเวิร์ก 1 เน็ตเวิร์ก

เราทดสอบโดยการนำข้อมูลที่อยู่ใน*ชุดสอน*มาสอนนิวรอลเน็ตเวิร์ก จากนั้นนำเน็ตเวิร์กที่ได้มาทดสอบกับ*ชุดทดสอบที่ 1* และ *ชุดทดสอบที่ 2* ได้ผลการทดลองดังที่แสดงในตารางที่ 4.1

ตารางที่ 4.1 ค่าความถูกต้องจากนิวรอลเน็ตเวิร์กเดี่ยว

ข้อมูลที่ใช้ทดสอบ	จำนวนตัวอย่างที่ผิด (เปอร์เซ็นต์ความถูกต้อง)
ชุดทดสอบที่ 1	115 (96.48%)
ชุดทดสอบที่ 2	112 (96.57%)

4.3.2 ทดสอบกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างโดยการแบ่งชุดข้อมูลสอนออกเป็น 6 ชุดตามรูปแบบของตัวอักษร

ผลการทดสอบกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างโดยการแบ่งชุดข้อมูลสอนออกเป็น 6 ชุด แล้วรวมผลลัพธ์ด้วยวิธีต่างๆ แสดงให้เห็นในตารางที่ 4.2 และตารางที่ 4.3

ตารางที่ 4.2 ผลการทดลองกับ*ชุดทดสอบที่ 1* ที่ได้จากเน็ตเวิร์กที่สร้างโดยการแบ่งชุดข้อมูลสอนตามรูปแบบของตัวอักษร แล้วนำผลลัพธ์มารวมกันด้วยวิธีต่างๆ

วิธีที่ใช้ในการรวมผลลัพธ์	**ชุดทดสอบที่ 1**	
	จำนวนตัวที่ผิด	**เปอร์เซ็นต์ความถูกต้อง**
MV	613	81.22%
MVP	244	92.52%
AV	242	92.59%
MMWV	244	92.52%
CWV	241	92.62%
HV (12 กลุ่ม)	209	93.60%
HV (9 กลุ่ม)	212	93.50%

ตารางที่ 4.3 ผลการทดลองกับ*ชุดทดสอบที่ 2* ที่ได้จากเน็ตเวิร์กที่สร้างโดยการแบ่งชุดข้อมูลสอนตามรูปแบบของตัวอักษร แล้วนำผลลัพธ์มารวมกันด้วยวิธีต่างๆ

วิธีที่ใช้ในการรวมผลลัพธ์	**ชุดทดสอบที่ 2**	
	จำนวนตัวที่ผิด	**เปอร์เซ็นต์ความถูกต้อง**
MV	581	82.20%
MVP	247	92.43%
AV	244	92.52%
MMWV	236	92.77%
CWV	244	92.52%
HV (12 กลุ่ม)	215	93.41%
HV (9 กลุ่ม)	207	93.66%

4.3.3 ทดสอบกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างด้วยวิธี Bagging

ผลการทดสอบกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างด้วยวิธี Bagging จำนวน 21 เน็ตเวิร์ก แล้วรวมผลลัพธ์ด้วยวิธีไม่ถ่วงน้ำหนัก และวิธีถ่วงน้ำหนัก ซึ่งได้แก่วิธี MV, MVP, AV, MMWV และ CWV ที่จำนวนเน็ตเวิร์กต่างๆกัน ผลที่ได้แสดงในตารางที่ 4.4 ตารางที่ 4.5 รูปที่ 4.1และรูปที่ 4.2

ตารางที่ 4.4 ผลการทดลองกับ*ชุดทดสอบที่ 1* ที่ได้จากเน็ตเวิร์กที่สร้างด้วยวิธี Bagging แล้วรวมผลลัพธ์ด้วยวิธีต่างๆ

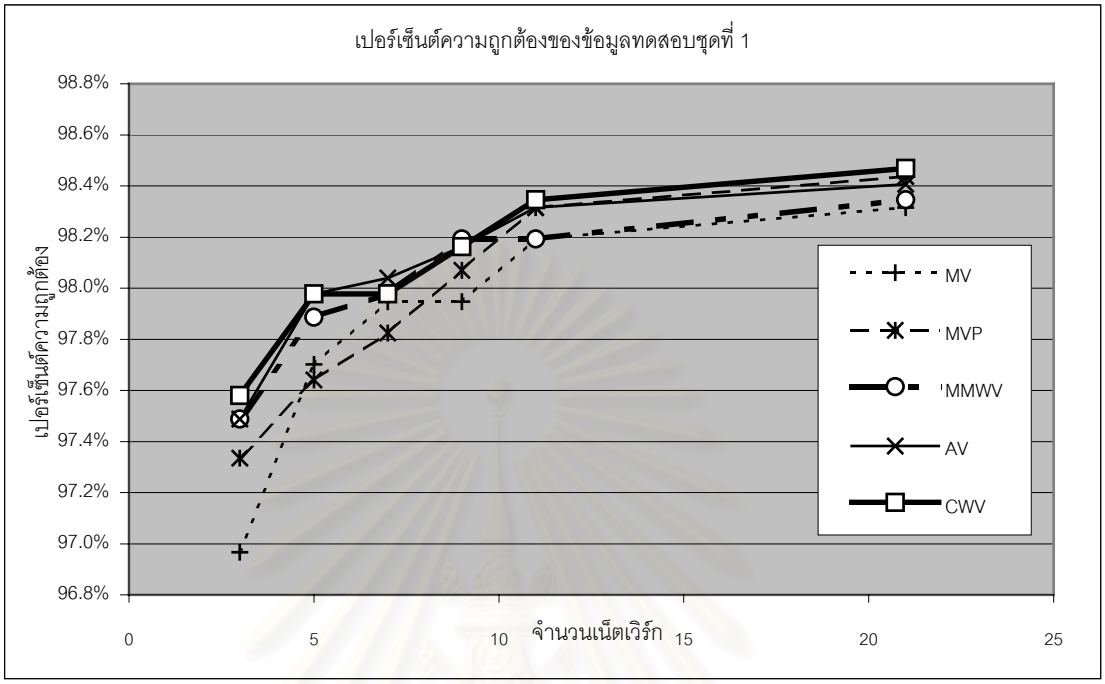
วิธีที่ใช้ในการรวมผลลัพธ์	**จำนวนเน็ตเวิร์ก**					
	3	5	7	9	11	21
MV	99 (96.97%)	75 (97.70%)	67 (97.95%)	67 (97.95%)	59 (98.19%)	55 (98.31%)
MVP	87 (97.33%)	77 (97.64%)	71 (97.82%)	63 (98.07%)	55 (98.31%)	51 (98.44%)
AV	82 (97.49%)	66 (97.98%)	64 (98.04%)	60 (98.16%)	55 (98.31%)	52 (98.41%)
MMWV	82 (97.49%)	69 (97.89%)	66 (97.98%)	59 (98.19%)	59 (98.19%)	54 (98.35%)
CWV	79 (97.58%)	66 (97.98%)	66 (97.98%)	60 (98.16%)	54 (98.35%)	50 (98.47%)

(ค่าตัวเลขคือจำนวนตัวอักษรที่ผิดจาก 3264 ตัวอักษร ค่าในวงเล็บคือเปอร์เซ็นต์ความถูกต้อง)

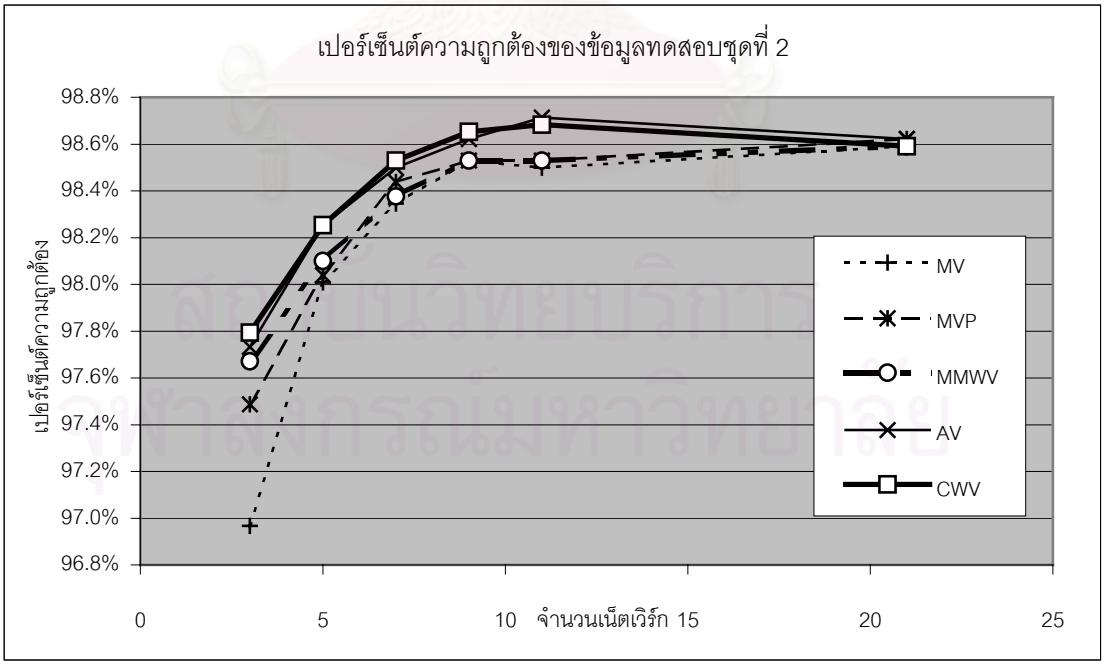
ตารางที่ 4.5 ผลการทดลองกับ*ชุดทดสอบที่ 2* ที่ได้จากเน็ตเวิร์กที่สร้างด้วยวิธี Bagging แล้วรวมผลลัพธ์ด้วยวิธีต่างๆ

วิธีที่ใช้ในการรวมผลลัพธ์	**จำนวนเน็ตเวิร์ก**					
	3	5	7	9	11	21
MV	99 (96.97%)	65 (98.01%)	54 (98.35%)	48 (98.53%)	49 (98.50%)	46 (98.59%)
MVP	82 (97.49%)	64 (98.04%)	51 (98.44%)	48 (98.53%)	48 (98.53%)	45 (98.62%)
AV	74 (97.73%)	57 (98.25%)	49 (98.50%)	45 (98.62%)	42 (98.71%)	45 (98.62%)
MMWV	76 (97.67%)	62 (98.10%)	53 (98.38%)	48 (98.53%)	48 (98.53%)	46 (98.59%)
CWV	72 (97.79%)	57 (98.25%)	48 (98.53%)	44 (98.65%)	43 (98.68%)	46 (98.59%)

(ค่าตัวเลขคือจำนวนตัวอักษรที่ผิดจาก 3264 ตัวอักษร ค่าในวงเล็บคือเปอร์เซ็นต์ความถูกต้อง)



รูปที่ 4.1 แสดงค่าความถูกต้องของ*ข้อมูลทดสอบชุดที่ 1* โดยวิธี MV MVP AV MMWV และ CWV



รูปที่ 4.2 แสดงค่าความถูกต้องของ*ข้อมูลทดสอบชุดที่ 2* โดยวิธี MV MVP AV MMWV และ CWV

ตารางที่ 4.6 ความผิดพลาดของกลุ่มก้อนของนิวรอลเน็ตเวิร์กแยกตามตัวอักษรบน *ชุดทดสอบที่ 1*

ตัวอักษร	จำนวน	วิธีที่ใช้ในการรวมผลลัพธ์				
		MV	MVP	AV	MMWV	CWV
ก	48	0	0	0	0	0
ข	48	3	3	3	3	2
ฃ	48	3	4	3	3	3
ค	48	6	6	6	6	6
ฅ	48	0	0	0	0	0
ฆ	48	0	0	0	0	0
ง	48	0	0	0	0	0
จ	48	1	1	1	1	1
ฉ	48	5	3	3	3	3
ช	48	6	6	6	6	6
ซ	48	6	6	6	5	6
ฌ	48	0	0	0	0	0
ญ	48	0	0	0	0	0
ฎ	48	14	11	11	10	11
ฏ	48	8	6	4	4	4
ฐ	48	0	0	0	0	0
ฑ	48	0	0	0	0	0
ฒ	48	0	0	0	0	0
ณ	48	0	0	0	0	0
ด	48	1	1	1	1	1
ต	48	1	0	0	0	0
ถ	48	1	1	0	0	0
ท	48	2	1	1	1	1
ธ	48	1	1	1	1	1
น	48	0	0	0	0	0
บ	48	0	0	0	0	0
ป	48	0	0	0	0	0
ผ	48	0	0	0	0	0
ฝ	48	0	0	0	0	0
พ	48	2	2	2	2	2
ฟ	48	2	1	1	1	1
ภ	48	1	0	0	0	0
ม	48	0	0	0	0	0
ย	48	0	0	0	1	0
ร	48	0	0	0	0	0

ตารางที่ 4.6 ความผิดพลาดของกลุ่มก้อนของนิวรอลเน็ตเวิร์กแยกตามตัวอักษรบน *ชุดทดสอบที่ 1* (ต่อ)

ตัวอักษร	จำนวน	วิธีที่ใช้ในการรวมผลลัพธ์				
		MV	MVP	AV	MMWV	CWV
ฤ	48	5	1	1	1	1
ล	48	1	0	0	0	0
ฦ	48	0	0	0	1	0
ว	48	1	1	1	1	1
ศ	48	1	1	1	1	1
ษ	48	1	1	1	2	1
ส	48	0	0	0	0	0
ห	48	0	0	0	0	0
ฬ	48	4	2	2	2	2
อ	48	0	0	0	0	0
ฮ	48	0	0	1	1	1
ฯ	48	1	1	1	1	1
ะ	48	0	0	0	0	0
ั	48	0	0	0	0	0
า	48	0	0	0	0	0
ิ	48	0	0	0	0	0
ี	48	1	1	1	1	1
ึ	48	3	3	4	3	2
ื	48	5	5	5	5	5
ุ	48	0	1	1	1	1
ู	48	1	0	0	0	0
เ	48	0	0	0	0	0
โ	48	4	4	3	2	3
ใ	48	3	3	3	3	3
ไ	48	2	4	3	3	3
ๆ	48	0	0	0	0	0
็	48	1	1	1	1	1
่	48	0	2	1	2	1
้	48	0	0	0	0	0
๊	48	0	0	0	0	0
๋	48	1	1	1	1	1
์	48	0	0	0	0	0
ำ	48	1	2	2	2	2
รวม	3264	99	87	82	82	79

ตารางที่ 4.7 ความผิดพลาดของกลุ่มก้อนของนิวรอลเน็ตเวิร์กแยกตามตัวอักษรบน *ชุดทดสอบที่ 2*

ตัวอักษร	จำนวน	วิธีที่ใช้ในการรวมผลลัพธ์				
		MV	MVP	AV	MMWV	CWV
ก	48	5	0	0	0	1
ข	48	2	1	1	1	1
ฃ	48	9	7	6	8	6
ค	48	4	3	2	2	2
ฅ	48	4	2	1	1	1
ฆ	48	0	0	0	0	0
ง	48	0	0	0	0	0
จ	48	1	2	1	1	1
ฉ	48	2	2	1	1	1
ช	48	5	6	6	5	6
ซ	48	5	7	6	6	6
ฌ	48	0	0	0	0	0
ญ	48	1	1	0	1	0
ฎ	48	13	12	13	12	11
ฏ	48	0	0	0	0	0
ฐ	48	0	0	0	0	0
ฑ	48	0	0	0	0	0
ฒ	48	0	0	0	0	0
ณ	48	0	0	0	0	0
ด	48	1	2	3	3	3
ต	48	6	6	4	4	4
ถ	48	1	1	1	1	1
ท	48	0	0	0	0	0
ธ	48	2	1	2	2	2
น	48	0	1	1	1	0
บ	48	2	1	1	1	1
ป	48	0	0	0	0	0
ผ	48	1	0	0	0	0
ฝ	48	0	0	0	0	0
พ	48	0	0	0	0	0
ฟ	48	0	0	0	0	0
ภ	48	0	0	0	0	0
ม	48	1	0	0	0	0
ย	48	3	1	0	0	0
ร	48	0	0	0	0	0

ตารางที่ 4.7 ความผิดพลาดของกลุ่มก้อนของนิวรอลเน็ตเวิร์กแยกตามตัวอักษรบน *ชุดทดสอบที่ 2*(ต่อ)

ตัวอักษร	จำนวน	วิธีที่ใช้ในการรวมผลลัพธ์				
		MV	MVP	AV	MMWV	CWV
ฤ	48	3	1	1	1	1
ล	48	0	0	0	0	0
ฦ	48	1	1	1	1	1
ว	48	0	0	0	0	0
ศ	48	0	0	0	0	0
ษ	48	0	0	0	0	0
ส	48	0	0	0	0	0
ห	48	0	0	0	0	0
ฬ	48	0	1	2	2	1
อ	48	2	2	2	2	2
ฮ	48	0	0	0	0	0
ฯ	48	4	3	2	3	2
ะ	48	1	0	0	0	0
ั	48	0	0	0	0	0
า	48	2	2	1	1	1
ิ	48	0	0	0	0	0
ี	48	2	2	2	2	2
ึ	48	1	1	1	1	1
ื	48	2	2	2	2	2
ุ	48	0	0	0	0	0
ู	48	0	0	0	0	0
เ	48	1	1	1	1	1
โ	48	4	4	4	4	4
ใ	48	3	2	2	2	3
ไ	48	2	2	2	2	2
ๆ	48	0	0	0	0	0
็	48	1	1	1	1	1
่	48	0	0	0	0	0
้	48	0	0	0	0	0
๊	48	1	0	0	0	0
๋	48	0	0	0	0	0
์	48	0	0	0	0	0
๎	48	1	1	1	1	1
รวม	3264	99	82	74	76	72

เมื่อพิจารณาความผิดพลาดของกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่รวมด้วยวิธี MV, MVP, MMWV, AV และ CWV โดยแยกตามตัวอักษร จากการทดลองจะพบว่าตัวอักษรบางตัว เช่น ฎ, ฆ, ช, โ, ไ และ ใ ยังคงมีความผิดพลาดสูง ซึ่งแสดงให้เห็นในตารางที่ 4.6 และ ตารางที่ 4.7 ตารางนี้ใช้ผลที่ได้จากกลุ่มก้อนของนิวรอลเน็ตเวิร์ก 3 เน็ตเวิร์ก สามารถดูรายละเอียดตัวที่อักษรที่ผิดและคำตอบที่ตอบผิดได้จากภาคผนวก ข และตัวอักษรที่ผิดโดยแยกตามรูปแบบและขนาดได้ที่ภาคผนวก ค

ปัญหาที่ตัวอักษรบางตัวยังคงมีความผิดพลาดสูงเนื่องจากเน็ตเวิร์กที่สร้างได้ยังคงมีความผิดพลาดซ้ำซ้อนกัน การที่จะแก้ไขข้อผิดพลาดนี้ได้สามารถทำได้ 2 วิธี คือ 1) สร้างเน็ตเวิร์กที่มีความผิดพลาดแตกต่างกัน 2) รวมผลลัพธ์ที่ได้ด้วยวิธีพิเศษ ซึ่งในที่นี้เราได้เลือกใช้วิธีที่ 2 คือ รวมผลลัพธ์ด้วยวิธี HV ซึ่งวิธี HV นี้เป็นวิธีที่คิดขึ้นมาเพื่อแก้ปัญหาในกรณีนี้

เราได้ทดลองรวมผลลัพธ์ด้วยวิธี HV โดยใช้กลุ่มของเน็ตเวิร์กเฉพาะทางที่เลือกด้วยมือ (12 กลุ่ม) เลือกอัตโนมัติโดยโปรแกรม (9 กลุ่ม) และใช้นิวรอลเน็ตเวิร์กในขั้นแรกจำนวน 3 เน็ตเวิร์ก ค่าความถูกต้องที่แสดงให้เห็นในตารางที่ 4.8 เป็นค่าที่ได้จากการกำหนดค่าขอบเขตบนขอบเขตล่าง และค่าขีดแบ่งที่ทำให้ได้ค่าความถูกต้องสูงสุด

ตารางที่ 4.8 ผลการทดลองกับ*ชุดทดสอบที่ 1* และ *ชุดทดสอบที่ 2* ที่ได้จากเน็ตเวิร์กที่สร้างด้วยวิธี Bagging แล้วรวมผลลัพธ์ด้วยวิธี HV เมื่อใช้จำนวนกลุ่มต่างๆ

วิธีที่ใช้ในการรวมผลลัพธ์	**ชุดทดสอบที่ 1**	**ชุดทดสอบที่ 2**
HV (12 กลุ่ม)	69 (97.89%)	62 (98.10%)
HV (9 กลุ่ม)	81 (97.52%)	66 (97.98%)

(ค่าตัวเลขคือจำนวนตัวอักษรที่ผิดจาก 3264 ตัวอักษร ค่าในวงเล็บคือเปอร์เซ็นต์ความถูกต้อง)

เมื่อพิจารณาความผิดพลาดของกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่รวมด้วยวิธี HV (12 กลุ่ม) และ HV (9 กลุ่ม) โดยแยกตามตัวอักษรแล้วจะพบว่าตัวอักษรบางตัวที่ยังคงมีความผิดพลาดสูง ซึ่งแสดงให้เห็นในตารางที่ 4.6 และ ตารางที่ 4.7 นั้นมีความผิดพลาดน้อยลง ความผิดพลาดแยกตามตัวอักษรของกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่รวมด้วยวิธี HV ได้แสดงให้เห็นในตารางที่ 4.9 ซึ่งตารางนี้ใช้ผลที่ได้จากกลุ่มก้อนของนิวรอลเน็ตเวิร์กในขั้นแรกเป็น 3 เน็ตเวิร์ก

ตารางที่ 4.9 ความผิดพลาดของกลุ่มก้อนของนิวรอลเน็ตเวิร์กแยกตามตัวอักษรบน*ชุดทดสอบ*

ตัวอักษร	จำนวน	ชุดทดสอบที่ 1		ชุดทดสอบที่ 2	
		HV12	HV9	HV12	HV9
ก	48	0	0	0	0
ข	48	2	2	1	1
ฃ	48	2	2	6	6
ค	48	5	6	1	2
ฅ	48	0	0	1	1
ฆ	48	0	0	0	0
ง	48	0	1	0	0
จ	48	1	0	1	1
ฉ	48	3	1	1	1
ช	48	5	5	4	4
ซ	48	6	8	5	5
ฌ	48	0	0	0	1
ญ	48	0	0	0	0
ฎ	48	8	12	10	11
ฏ	48	5	4	0	0
ฐ	48	0	0	0	0
ฑ	48	0	0	0	0
ฒ	48	0	1	0	0
ณ	48	0	0	0	0
ด	48	1	1	2	3
ต	48	0	0	3	3
ถ	48	0	0	1	1
ท	48	1	1	0	0
ธ	48	1	1	1	1
น	48	0	0	1	1
บ	48	0	0	1	1
ป	48	0	0	0	0
ผ	48	0	0	0	0
ฝ	48	0	0	0	0
พ	48	2	2	0	0
ฟ	48	1	1	0	0
ภ	48	0	0	0	0
ม	48	0	0	0	0
ย	48	1	0	0	0
ร	48	0	0	0	0

ตารางที่ 4.9 ความผิดพลาดของกลุ่มก้อนของนิวรอลเน็ตเวิร์กแยกตามตัวอักษรบน*ชุดทดสอบ* (ต่อ)

ตัวอักษร	จำนวน	ชุดทดสอบที่ 1		ชุดทดสอบที่ 2	
		HV12	HV9	HV12	HV9
ฤ	48	1	3	1	1
ล	48	0	0	0	0
ฦ	48	0	1	1	1
ว	48	1	3	0	0
ศ	48	1	1	0	0
ษ	48	1	1	0	0
ส	48	0	0	0	0
ห	48	0	0	0	0
ฟ	48	3	2	1	1
อ	48	0	0	2	1
ฮ	48	0	1	0	0
ฯ	48	1	1	3	2
ะ	48	0	0	0	0
ั	48	0	0	0	0
า	48	0	0	1	3
ิ	48	0	0	0	0
ี	48	1	1	2	2
ึ	48	2	3	0	2
ื	48	3	2	3	1
ุ	48	1	1	0	0
ู	48	0	0	0	0
เ	48	0	0	1	0
โ	48	2	5	2	3
ใ	48	1	1	2	2
ไ	48	2	3	2	2
ๆ	48	0	0	0	0
็	48	1	1	1	1
่	48	1	1	0	0
้	48	0	0	0	0
๊	48	0	0	0	0
๋	48	1	1	0	0
์	48	0	0	0	0
ำ	48	2	1	1	1
รวม	3264	69	81	62	66

4.4 วิเคราะห์ผลการทดลอง

4.4.1 กลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างโดยการแบ่งชุดข้อมูลสอนออกเป็น 6 ชุดตามรูปแบบของตัวอักษร

กลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างด้วยวิธีนี้ให้ความถูกต้องต่ำกว่าการใช้นิวรอลเน็ตเวิร์กเดี่ยวเนื่องจากการแบ่งข้อมูลสอนออกเป็น 6 ชุด ทำให้จำนวนข้อมูลที่ใช้สอนนิวรอลเน็ตเวิร์กแต่ละเน็ตเวิร์กมีจำนวนน้อยลง ซึ่งแต่ละชุดจะประกอบด้วยข้อมูลเพียง 544 ตัว ทำให้เน็ตเวิร์กได้เรียนตัวอักษรแต่ละตัวที่ขนาดต่างๆกัน เพียงแค่ 8 ตัว การรู้จำจึงทำได้ไม่ดีเท่าที่ควร

4.4.2 กลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างด้วยวิธี Bagging

ผลการทดลองเปรียบเทียบความถูกต้องของนิวรอลเน็ตเวิร์กเดี่ยว (ตารางที่ 4.1) กับความถูกต้องของกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างด้วยวิธี Bagging (ตารางที่ 4.4 และตารางที่ 4.5) แสดงให้เห็นว่าการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์กทั้ง 6 วิธีให้ความถูกต้องมากขึ้น ซึ่งแสดงถึงประสิทธิภาพของกลุ่มก้อนที่เหนือกว่านิวรอลเน็ตเวิร์กเดี่ยว ดูการเปรียบเทียบได้จากตารางที่ 4.10 ตารางที่ 4.11 ตารางที่ 4.12 ตารางที่ 4.13 ตารางที่ 4.14 ตารางที่ 4.15 รูปที่ 4.3 และรูปที่ 4.4

- วิธีไม่ถ่วงน้ำหนัก ซึ่งได้แก่ วิธี MV, MVP และ AV

สำหรับวิธีไม่ถ่วงน้ำหนัก วิธี AV ให้ความถูกต้องสูงสุดทั้งใน *ชุดทดสอบที่ 1* และ *ชุดทดสอบที่ 2* วิธีการ MVP ให้ผลใกล้เคียงกับ MV โดยให้ความถูกต้องที่สูงกว่าเล็กน้อย

สาเหตุที่วิธี AV ให้ผลที่มีความถูกต้องสูงกว่าวิธี MV และ MVP เนื่องจากวิธี AV นั้นใช้ค่าเอาต์พุตทุกค่าที่ได้จากโนดทุกโนดของนิวรอลเน็ตเวิร์ก ขณะที่วิธี MV และ MVP ใช้เพียงค่าสูงสุดเพียงค่าเดียว ซึ่งเป็นการละทิ้งข้อมูลที่ได้จากโนดที่สำคัญบางโนดไป เพราะคำตอบที่ได้จากนิวรอลเน็ตเวิร์กในบางครั้งก็มีคำตอบที่ให้ค่าสูง (มากกว่า 0.5) มากกว่า 1 ค่า ซึ่งในกรณีนี้การใช้ค่าสูงสุดเพียงค่าเดียวจึงไม่ใช่วิธีที่เหมาะสม

ส่วนวิธี MVP นั้นให้ความถูกต้องสูงกว่าวิธี MV เนื่องจากวิธี MVP นั้น ใช้ค่าเอาต์พุตที่ได้จากโนดของนิวรอลเน็ตเวิร์กในการหาคำตอบ ขณะที่วิธี MV ใช้เพียงแค่คำตอบที่ได้โดยไม่สนใจค่าเอาต์พุต จึงเป็นการละทิ้งข้อมูลบางส่วนเช่นกัน และเนื่องจากการใช้เพียงค่าคำตอบทำให้ไม่สามารถบอกได้ว่าเน็ตเวิร์กนั้นมีความมั่นใจกับคำตอบนั้นๆ

แค่ไหน จึงเหมือนกับว่าทุกคำตอบมีความสำคัญเท่ากัน ซึ่งที่จริงแล้วแต่ละคำตอบมีความสำคัญไม่เท่ากัน

อย่างไรก็ตามทุกวิธีให้ความถูกต้องสูงกว่าการใช้นิวรอลเน็ตเวิร์กเดี่ยว ดูการเปรียบเทียบได้จากตารางที่ 4.10 ตารางที่ 4.11

ตารางที่ 4.10 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก บน *ชุดทดสอบที่ 1* เมื่อเทียบกับการใช้นิวรอลเน็ตเวิร์กเดี่ยว

วิธีที่ใช้ในการรวมผลลัพธ์	**จำนวนเน็ตเวิร์ก**					
	3	5	7	9	11	21
MV	13.91%	34.78%	41.74%	41.74%	48.70%	52.17%
MVP	24.35%	33.04%	38.26%	45.22%	52.17%	55.65%
AV	28.70%	42.61%	44.35%	47.83%	52.17%	54.78%

ตารางที่ 4.11 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก บน *ชุดทดสอบที่ 2* เมื่อเทียบกับการใช้นิวรอลเน็ตเวิร์กเดี่ยว

วิธีที่ใช้ในการรวมผลลัพธ์	**จำนวนเน็ตเวิร์ก**					
	3	5	7	9	11	21
MV	11.61%	41.96%	51.79%	57.14%	56.25%	58.93%
MVP	26.79%	42.86%	54.46%	57.14%	57.14%	59.82%
AV	33.93%	49.11%	56.25%	59.82%	62.50%	59.82%

- วิธีถ่วงน้ำหนัก ซึ่งได้แก่ วิธี MMWV และ CWV

สำหรับวิธี MMWV นั้นโดยรวมแล้วมีประสิทธิภาพดีกว่าวิธี MV และ MVP ซึ่งใช้เพียงคำตอบ หรือเอาต์พุตที่ได้จากโนดที่มีค่ามากที่สุดเพียงโนดเดียว ซึ่งวิธี MMWV ก็ใช้เพียงเอาต์พุตที่ได้จากโนดที่มีค่ามากที่สุดเพียงโนดเดียว แต่มีการเพิ่มน้ำหนักให้กับนิวรอลเน็ตเวิร์กแต่ละเน็ตเวิร์ก อย่างไรก็ตามเมื่อเทียบกับวิธี AV ซึ่งใช้ค่าเอาต์พุตที่ได้จากทุกโนดของทุกเน็ตเวิร์กแล้วจะพบว่า โดยรวมแล้ววิธี MMWV จะมีประสิทธิภาพด้อยกว่าเล็กน้อย แต่จะเห็นได้ชัดว่าค่าความแตกต่างระหว่างค่าเอาต์พุตของนิวรอลเน็ตเวิร์กที่มีค่ามากที่สุดกับค่าเอาต์พุตที่มีค่ามากรองลงมา สามารถบอกได้ว่าเน็ตเวิร์กนั้นๆมีความน่าเชื่อถือเพียงใด โดยหากความแตกต่างดังกล่าวมีค่ามากแสดงว่าเน็ตเวิร์กค่อนข้างมั่นใจคำตอบจึงกำหนดให้ความน่าเชื่อถือมาก และในทางตรงกันข้ามหากความแตกต่างมีค่าน้อย ซึ่งอาจเกิดจากเอาต์พุตมีค่าสูงมากกว่า 1 ค่า หรือเอาต์พุตไม่มีค่าสูงเลยมีแต่ค่าต่ำ

ซึ่งที่จริงแล้วคำตอบที่ดีควรจะมีโนดที่มีค่าสูงโนดเดียวและโนดที่เหลือเป็นค่าต่ำหมด แต่ในการทำงานจริงบางครั้งก็เกิดเหตุการณ์ดังกล่าว ซึ่งแสดงให้เห็นว่าเน็ตเวิร์กไม่มั่นใจคำตอบ ดังนั้นความน่าเชื่อถือของเน็ตเวิร์กนั้นก็จะน้อย

ส่วนวิธี CWV นั้นโดยรวมแล้วมีประสิทธิภาพดีกว่าวิธี MV MVP และ MMWV มาก แต่เมื่อเทียบกับวิธี AV จะพบว่าวิธี CWV จะมีประสิทธิภาพดีกว่าเล็กน้อย ดังนั้นจะเห็นว่าการใช้ค่าน้ำหนักที่คำนวณจากสัดส่วนของจำนวนครั้งที่เน็ตเวิร์กนั้นได้เรียนรู้คลาสของข้อมูลสามารถบอกได้ว่าเน็ตเวิร์กนั้นๆมีความน่าเชื่อถือเพียงใด โดยหากเน็ตเวิร์กได้เรียนรู้คลาสนั้นๆมาก ความน่าเชื่อถือของเน็ตเวิร์กก็จะมากตามไปด้วย

ผลที่ได้แสดงให้เห็นว่าวิธีที่ใช้ค่าน้ำหนักทั้งวิธี MMWV และวิธี CWV ช่วยเพิ่มประสิทธิภาพของการรู้จำตัวพิมพ์อักษรไทยได้ เมื่อเทียบกับการรวมโดยวิธีไม่ใช้ค่าน้ำหนัก (MV MVP และ AV)

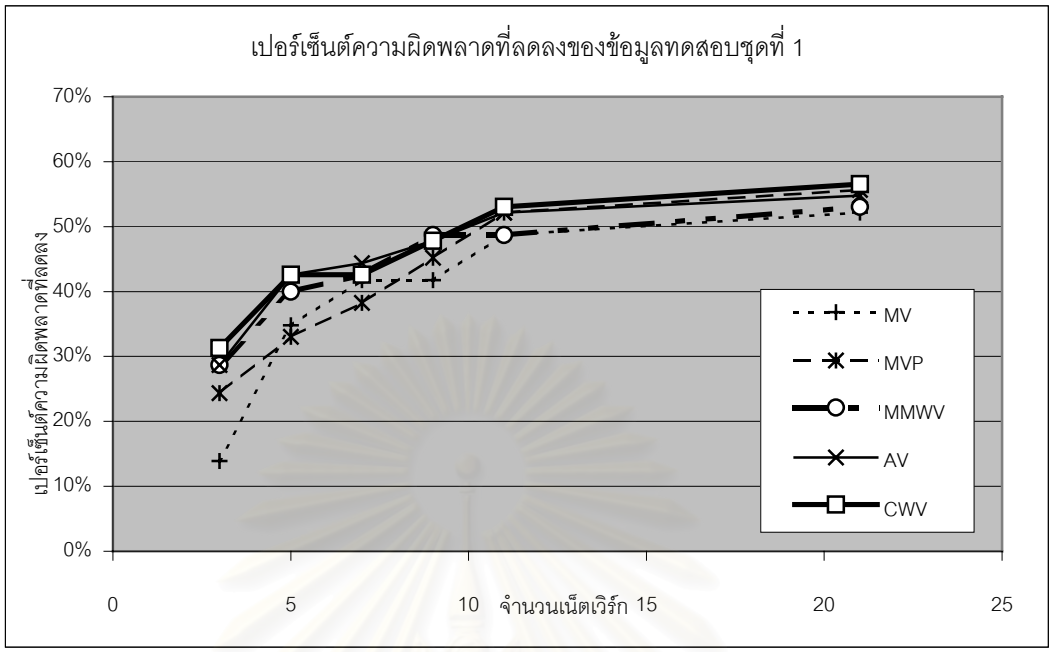
เมื่อเปรียบเทียบผลของแต่ละวิธีแล้วจะเห็นว่าวิธี CWV ให้ความถูกต้องสูงสุด และทุกวิธีก็ให้ความถูกต้องสูงกว่าการใช้นิวรอลเน็ตเวิร์กเดี่ยว ดูการเปรียบเทียบได้จากตารางที่ 4.12 และ ตารางที่ 4.13

ตารางที่ 4.12 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก บน *ชุดทดสอบที่ 1* เมื่อเทียบกับการใช้นิวรอลเน็ตเวิร์กเดี่ยว

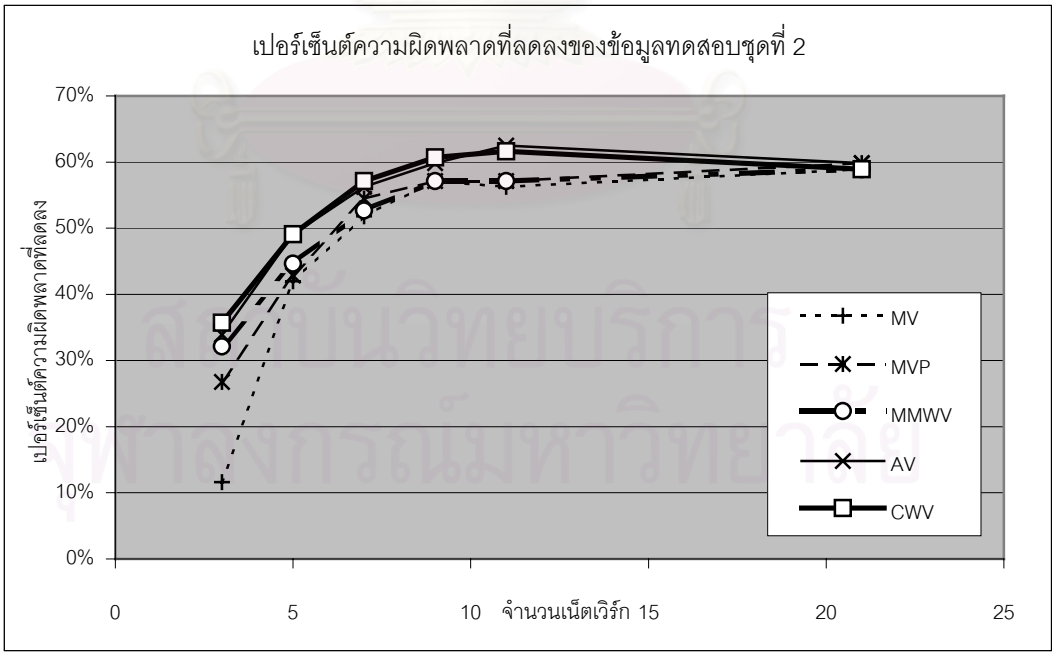
วิธีที่ใช้ในการรวมผลลัพธ์	**จำนวนเน็ตเวิร์ก**					
	3	5	7	9	11	21
MMWV	28.70%	40.00%	42.61%	48.70%	48.70%	53.04%
CWV	31.30%	42.61%	42.61%	47.83%	53.04%	56.52%

ตารางที่ 4.13 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก บน *ชุดทดสอบที่ 2* เมื่อเทียบกับการใช้นิวรอลเน็ตเวิร์กเดี่ยว

วิธีที่ใช้ในการรวมผลลัพธ์	**จำนวนเน็ตเวิร์ก**					
	3	5	7	9	11	21
MMWV	32.14%	44.64%	52.68%	57.14%	57.14%	58.93%
CWV	35.71%	49.11%	57.14%	60.71%	61.61%	58.93%



รูปที่ 4.3 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก โดยวิธี MV, MVP, AV, MMWV และ CWV บน *ชุดทดสอบที่ 1* เมื่อเทียบกับการใช้นิวรอลเน็ตเวิร์กเดี่ยว



รูปที่ 4.4 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์ก โดยวิธี MV, MVP, AV, MMWV และ CWV บน *ชุดทดสอบที่ 2* เมื่อเทียบกับการใช้นิวรอลเน็ตเวิร์กเดี่ยว

- วิธีเฉพาะปัญหา ซึ่งได้แก่ วิธี HV

สำหรับวิธีเฉพาะปัญหานั้น วิธี HV ทั้งวิธีที่เลือกด้วยมือ และเลือกอัตโนมัติโดยโปรแกรม ให้ผลที่ดีมากเมื่อเทียบกับวิธีอื่นๆ ที่จำนวนเน็ตเวิร์กเป็น 3 เน็ตเวิร์ก โดยเฉพาะวิธีเลือกด้วยมือที่ได้จำนวนกลุ่มเป็น 12 กลุ่มนั้นสามารถลดความผิดพลาดได้อย่างมาก ซึ่งวิธี HV นี้สามารถแยกแยะตัวอักษรได้ดีกว่าวิธีอื่นเนื่องจาก เมื่อสามารถกำหนดกลุ่มของคำตอบได้ถูกต้องเน็ตเวิร์กเฉพาะทางที่เรียนรู้เฉพาะตัวอักษรในกลุ่มนั้นๆจะสามารถแยกแยะตัวอักษรในกลุ่มได้ดีกว่าการใช้เน็ตเวิร์กที่สร้างด้วยวิธี Bagging

แต่เมื่อทดลองเพิ่มจำนวนเน็ตเวิร์กก็พบว่าวิธี HV อาจมีประสิทธิภาพด้อยกว่าวิธีอื่นโดยมีสาเหตุจากการที่วิธี HV ใช้คำตอบจากเน็ตเวิร์กเฉพาะทางเพียงเน็ตเวิร์กเดียวในการตอบ ขณะที่วิธีอื่นๆ ใช้ผลจากทุกเน็ตเวิร์กในหาการคำตอบ ซึ่งจากทฤษฎีในบทที่ 2 ได้แสดงให้เห็นแล้วว่าการใช้กลุ่มก้อนของเน็ตเวิร์กเมื่อเน็ตเวิร์กมีค่าความผิดพลาดไม่เกิน 0.5 ย่อมได้ผลที่ดีกว่าการใช้เน็ตเวิร์กเพียงเน็ตเวิร์กเดียว อย่างไรก็ตามในการใช้งานจริงเราไม่ต้องการใช้เน็ตเวิร์กจำนวนมาก เนื่องจากการใช้จำนวนเน็ตเวิร์กมากเกินไป เช่น 21 เน็ตเวิร์ก จะทำให้เสียเวลาในการคำนวณ ดังนั้นในกรณีที่ใช้จำนวนเน็ตเวิร์กไม่มากเช่น 3 เน็ตเวิร์ก วิธี HV นี้สามารถช่วยลดความผิดพลาดได้มากเมื่อเทียบกับนิวรอลเน็ตเวิร์กเดี่ยวหรือการรวมผลลัพธ์โดยวิธีอื่น ซึ่งแสดงให้เห็นในตารางที่ 4.14, ตารางที่ 4.15

ตารางที่ 4.14 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของการรวมผลลัพธ์ด้วยวิธี HV บน *ชุดทดสอบที่ 1* เมื่อเทียบกับการรวมผลลัพธ์ด้วยวิธีอื่นๆ

วิธีที่ใช้ในการรวมผลลัพธ์	**เน็ตเวิร์กเดี่ยว**	MV	MVP	AV	MMWV	CWV
HV(12 กลุ่ม)	40.00%	28.28%	18.39%	13.41%	13.41%	10.13%
HV(9 กลุ่ม)	29.57%	20.20%	9.20%	3.66%	3.66%	0.00%

ตารางที่ 4.15 เปอร์เซ็นต์ความผิดพลาดที่ลดลงของของการรวมผลลัพธ์ด้วยวิธี HV บน *ชุดทดสอบที่ 2* เมื่อเทียบกับการรวมผลลัพธ์ด้วยวิธีอื่นๆ

วิธีที่ใช้ในการรวมผลลัพธ์	**เน็ตเวิร์กเดี่ยว**	MV	MVP	AV	MMWV	CWV
HV(12 กลุ่ม)	44.64%	30.30%	15.85%	6.76%	9.21%	4.17%
HV(9 กลุ่ม)	41.07%	28.28%	13.41%	4.05%	6.58%	1.39%

จากการวิเคราะห์ตารางที่ 4.9 จะเห็นว่าวิธี HV สามารถลดความผิดพลาดในกรณีที่ตัวอักษรมีความคล้ายคลึงกันได้ เช่น ฎ, ช, โ, และ ใ และเมื่อเปรียบเทียบผลของแต่ละวิธีแล้วจะเห็นว่าวิธี HV นั้นเมื่อสามารถแบ่งกลุ่ม และหาค่าขอบเขตบน ขอบเขตล่าง และค่าขีดแบ่งที่เหมาะสมได้จะมีประสิทธิภาพที่ดีกว่าวิธี AV และนิวรอลเน็ตเวิร์กเดี่ยวมากเมื่อใช้จำนวนเน็ตเวิร์กน้อยๆ

4.5 **สรุปผลการทดลอง**

โดยสรุปแล้วการสร้างกลุ่มก้อนของนิวรอลเน็ตเวิร์กเพื่อใช้ในการรู้จำตัวอักษรพิมพ์ภาษาไทยนั้นเมื่อสร้างกลุ่มก้อนของนิวรอลเน็ตเวิร์กโดยการแบ่งชุดข้อมูลสอนออกเป็น 6 ชุดตามรูปแบบของตัวอักษรจะได้ความถูกต้องต่ำกว่าการใช้นิวรอลเน็ตเวิร์กเดี่ยว เนื่องจากจำนวนข้อมูลใน*ชุดสอน*มีน้อยกว่า ทำให้ประสิทธิภาพของนิวรอลเน็ตเวิร์กแต่ละเน็ตเวิร์กลดลงมากเมื่อเทียบกับนิวรอลเน็ตเวิร์กเดี่ยว แต่เมื่อพิจารณาดูถึงวิธีการในการรวมผลลัพธ์แล้วจะเห็นได้ชัดว่า วิธีการที่นำเสนอ คือวิธี MMWV, CWV และ HVนั้น โดยรวมแล้วให้ความผิดพลาดลดลง

แต่ในกรณีที่สร้างกลุ่มก้อนของนิวรอลเน็ตเวิร์กโดยวิธี Bagging เมื่อพิจารณาความสัมพันธ์ของจำนวนเน็ตเวิร์กกับเปอร์เซ็นต์ความถูกต้องจะเห็นว่า จำนวนของเน็ตเวิร์กมีผลต่อเปอร์เซ็นต์ความถูกต้องของการรู้จำตัวอักษรพิมพ์ภาษาไทย โดยที่เมื่อใช้จำนวนเน็ตเวิร์กเพิ่มมากขึ้นเปอร์เซ็นต์ความถูกต้องในการรู้จำก็จะเพิ่มขึ้น แต่อัตราการเพิ่มขึ้นของเปอร์เซ็นต์ความถูกต้องจะน้อยลงเมื่อจำนวนเน็ตเวิร์กเพิ่มขึ้น ซึ่งมีสาเหตุมาจากการที่เน็ตเวิร์กที่ได้จากการสร้างด้วยวิธี Bagging ยังคงให้ผลที่ผิดซ้ำซ้อนกัน

เมื่อพิจารณาความสัมพันธ์ระหว่างจำนวนเน็ตเวิร์กกับวิธีที่ใช้ในการรวมผลลัพธ์จะเห็นว่าที่จำนวนเน็ตเวิร์กน้อยๆวิธีที่ใช้ในการรวมผลลัพธ์ที่แตกต่างกันทำให้ได้เปอร์เซ็นต์ความถูกต้องที่ต่างกันมาก แต่เมื่อเพิ่มจำนวนเน็ตเวิร์กให้มากขึ้น วิธีที่ใช้ในการรวมผลลัพธ์ที่ต่างกันจะให้เปอร์เซ็นต์ความถูกต้องที่ไม่ต่างกันมากนัก

นอกจากนี้ยังเห็นได้ชัดว่ากลุ่มก้อนของนิวรอลเน็ตเวิร์กสามารถลดความผิดพลาดของการรู้จำตัวอักษรพิมพ์ภาษาไทยได้มาก โดยสามารถลดได้มากที่สุดถึง 56.52 เปอร์เซ็นต์ บน*ชุดทดสอบที่ 1* และ 62.50 เปอร์เซ็นต์ บน*ชุดทดสอบที่ 2* ซึ่งเมื่อพิจารณาดูถึงวิธีการในการรวมผลลัพธ์แล้วจะเห็นว่า ทุกวิธีการที่ใช้ในการทดลองให้ผลลัพธ์ที่ดีกว่าการใช้นิวรอลเน็ตเวิร์กเดี่ยว

โดยเฉพาะวิธีการที่นำเสนอ คือวิธี MMWV, CWV และ HV ที่จำนวนเน็ตเวิร์กน้อยๆนั้นให้ความผิดพลาดลดลงมาก ซึ่งในการใช้งานจริงเราก็ไม่ต้องการใช้เน็ตเวิร์กจำนวนมากเนื่องจากทำให้เสียเวลา ดังนั้นวิธีดังกล่าวจึงมีประโยชน์ในการเพิ่มความถูกต้องของการรู้จำตัวอักษรพิมพ์ภาษาไทย

4.6 ปัญหาและข้อจำกัด

1. ปัญหาเรื่องความเร็ว

   - ความเร็วในการสอนนิวรอลเน็ตเวิร์ก

   การใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์กนั้นให้ความถูกต้องที่มากกว่าการใช้นิวรอลเน็ตเวิร์กเดี่ยว แต่การสอนนิวรอลเน็ตเวิร์กเป็นจำนวนมากนั้นต้องใช้เวลามากขึ้น

   - ความเร็วในการหาคำตอบ

   การคิดด้วยวิธีที่ซับซ้อนขึ้นอย่างวิธี MMWV ที่ต้องหาค่าความแตกต่างระหว่างค่าเอาต์พุตของนิวรอลเน็ตเวิร์กที่มีค่ามากที่สุดกับค่าเอาต์พุตที่มีค่ามากรองลงมา วิธี CWV ที่ต้องนำค่าน้ำหนักมาคำนวณด้วย และ วิธี HV ที่ต้องใช้เน็ตเวิร์กเฉพาะทาง ทำให้ต้องใช้เวลาในการหาคำตอบของกลุ่มก้อนของนิวรอลเน็ตเวิร์กมากขึ้นกว่าวิธีพื้นฐาน

2. ปัญหาเรื่องหน่วยความจำ

เนื่องจากวิธีที่เสนอนั้นมีวิธีการคิดที่ซับซ้อนขึ้นทำให้ต้องใช้หน่วยความจำเป็นจำนวนมาก ทำให้ต้องการเครื่องที่มีสมรรถนะสูง หรือหากต้องการให้ใช้งานได้สำหรับเครื่องที่มีสมรรถนะต่ำจำเป็นจะต้องใช้การเก็บข้อมูลลงในแฟ้มข้อมูลซึ่งทำให้มีผลต่อความเร็วในการทำงาน

3. ปัญหาเรื่องการสร้างเน็ตเวิร์กที่จะนำมาใช้

การสร้างกลุ่มก้อนของนิวรอลเน็ตเวิร์กด้วยวิธี Bagging นั้น เน็ตเวิร์กที่ได้ยังไม่มีความหลากหลายในการให้คำตอบที่ผิด คือยังคงมีการผิดที่ซ้ำๆกัน ทำให้คำตอบรวมที่ได้จากกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่เกิดขึ้นจากกรณีนี้ไม่อาจแก้ไขให้ดีขึ้นได้

4. ปัญหาเรื่องการกำหนดค่าต่างๆ

- การกำหนดค่าต่างๆของวิธี HV

วิธี HV นั้นสามารถลดความผิดพลาดได้มากเมื่อเทียบกับวิธีอื่น โดยเฉพาะอย่างยิ่งเมื่อใช้จำนวนเน็ตเวิร์กน้อยๆ แต่หากกำหนดค่าขอบเขตบน ขอบเขตล่างและค่าขีดแบ่งไม่เหมาะสม จะทำให้ค่าความถูกต้องของวิธีนี้จะแย่กว่าวิธีอื่นๆ ดังนั้นการที่จะให้ได้ผลที่ดีนั้นต้องมีการเปลี่ยนค่าตัวแปรทั้ง 3 ค่านี้ให้เหมาะสมกับชุดข้อมูลและกลุ่มของเน็ตเวิร์กเฉพาะทาง ซึ่งในการกำหนดค่าเหล่านี้ไม่มีกฎเกณฑ์ที่แน่นอน จึงยากที่จะหาค่าที่ทำให้ได้ค่าเปอร์เซ็นต์ความถูกต้องที่สูงที่สุดได้

- การแบ่งกลุ่มของเน็ตเวิร์กเฉพาะทาง

การแบ่งกลุ่มของเน็ตเวิร์กเฉพาะทางเนื่องจากไม่มีกฎเกณฑ์ที่แน่นอน ดังนั้นการแบ่งกลุ่มให้เหมาะสมจึงเป็นเรื่องที่ทำได้ยาก

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

การรู้จำตัวพิมพ์อักษรไทยโดยการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์กนั้นมีขั้นตอนที่สำคัญอยู่ 2 ขั้นตอน คือ การสร้างกลุ่มก้อนของนิวรอลเน็ตเวิร์ก และการรวมผลลัพธ์ที่ได้จากกลุ่มก้อนของเน็ตเวิร์กเข้าด้วยกันเพื่อหาผลลัพธ์สุดท้าย ซึ่งจากงานวิจัยอื่นๆได้แสดงให้เห็นว่าวิธี AdaBoost เป็นวิธีที่มีความถูกต้องสูง หากแต่มีข้อเสียคือวิธีนี้ไม่สามารถจะทำงานได้ในกรณีที่ตัวแยกแยะไม่มีความผิดพลาดเลย ซึ่งในกรณีของข้อมูลชุดนี้ การใช้วิธี AdaBoost ในการสร้างนิวรอลเน็ตเวิร์กก็จะให้ผลไม่ต่างจากการใช้วิธี Bagging ดังนั้นงานวิจัยนี้จึงได้นำเสนอวิธีการรวมผลลัพธ์แบบใหม่ซึ่งสามารถใช้เพิ่มความถูกต้องของวิธี Bagging ซึ่งใช้วิธีรวมผลลัพธ์แบบ AV ได้

การทดลองใช้ผลการรู้จำจากตัวอย่างอักษร 2 ชุด ซึ่งได้จากการนำตัวอักษรต้นแบบไปถ่ายเอกสารให้จางลงและเข้มขึ้น แต่ละชุดประกอบด้วยตัวอักษรแบบ AngsanaUPC, BrowaliaUPC, CordiaUPC, DilleniaUPC, EucrosiaUPC และ FreesiaUPC โดยตัวอักษรแต่ละแบบจะประกอบด้วยตัวอักษรขนาด 16, 18, 20, 22, 24, 26, 28 และ 36 จุด อ่านผ่านเครื่องแสกนเนอร์ที่ความละเอียด 200 จุดต่อนิ้ว รวมเป็นจำนวนตัวอักษร 3264 ตัวต่อตัวอย่าง 1 ชุด หรือทั้งหมด 6528 ตัวอักษร

ในการทดลองได้ทดลองรวมผลลัพธ์ที่ได้จากกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างโดยการแบ่งชุดข้อมูลสอนออกเป็น 6 ชุดตามรูปแบบของตัวอักษร และจากกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างโดยวิธี Bagging

สำหรับกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างโดยการแบ่งชุดข้อมูลสอนออกเป็น 6 ชุดตามรูปแบบของตัวอักษรนั้นให้เปอร์เซ็นต์ความถูกต้องที่ต่ำกว่าการใช้นิวรอลเน็ตเวิร์กเดี่ยวไม่ว่าจะรวมด้วยวิธีใดก็ตาม

สำหรับกลุ่มก้อนของนิวรอลเน็ตเวิร์กที่สร้างโดยวิธี Bagging ได้ทดลองรวมผลลัพธ์ที่ได้จากกลุ่มก้อนของนิวรอลเน็ตเวิร์กด้วยวิธี MV, MVP, AV, MMWV, CWV และ HV ที่จำนวนเน็ตเวิร์กต่างๆกันเพื่อดูความสัมพันธ์ของจำนวนเน็ตเวิร์กต่อความถูกต้องและความสัมพันธ์ของวิธีที่ใช้รวมผลลัพธ์ต่อความถูกต้อง ผลที่ได้สามารถสรุปได้ดังนี้

- จำนวนของเน็ตเวิร์กมีผลต่อความถูกต้องของการรู้จำตัวอักษรพิมพ์ภาษาไทย โดยที่เมื่อใช้จำนวนเน็ตเวิร์กเพิ่มมากขึ้นเปอร์เซ็นต์ความถูกต้องในการรู้จำก็จะเพิ่มขึ้น แต่อัตราการเพิ่มขึ้นของเปอร์เซ็นต์ความถูกต้องจะน้อยลงเมื่อจำนวนเน็ตเวิร์กเพิ่มขึ้น เนื่องจากเน็ตเวิร์กที่ได้จากการสร้างด้วยวิธี Bagging ให้ผลที่ผิดซ้ำซ้อนกัน

- วิธีที่ใช้รวมผลลัพธ์ที่แตกต่างกันทำให้ได้ผลที่ต่างกัน โดยวิธีที่นำเสนอให้ผลดีที่กว่าวิธีพื้นฐานบนชุดทดสอบทั้ง 2 ชุด ที่จำนวนเน็ตเวิร์กน้อยๆ

โดยสรุปแล้วพบว่าการใช้กลุ่มก้อนของนิวรอลเน็ตเวิร์กช่วยเพิ่มความถูกต้องให้กับการรู้จำตัวพิมพ์อักษรไทยได้เป็นอย่างดี

5.2 **ข้อเสนอแนะ**

1. การใช้จำนวนตัวอักษรต้นแบบมากกว่าที่ใช้ในการวิจัยนี้ (การวิจัยนี้ใช้จำนวน 3264 ตัวอักษร) ในการเรียนรู้เพื่อให้การเรียนรู้ทำได้ดีขึ้น

2. การพัฒนาวิธีการเพิ่มเติมในกรณีที่ผลลัพธ์มีค่าสูง (มากกว่า 0.5) หลายโนด ซึ่งอาจจะทำได้โดยการรู้จำด้วยวิธีอื่นๆหรือหาค่าคุณสมบัติอื่นๆเพิ่มเติมเพื่อใช้ในการแยกแยะตัวอักษรนั้นๆได้ดีขึ้น

3. ปรับปรุงการสร้างเน็ตเวิร์กเพื่อให้เน็ตเวิร์กที่ได้มีค่าความผิดพลาดที่แตกต่างกัน เนื่องจากเน็ตเวิร์กที่ได้ในงานวิจัยนี้ยังคงมีการผิดที่ซ้ำซ้อนกันอยู่มาก

4. สำหรับวิธี CWV

   - ปรับปรุงขั้นตอนวิธีในการหาผลลัพธ์ของกลุ่มก้อนเพื่อให้ทำงานได้เร็วยิ่งขึ้น

5. สำหรับวิธี HV

   - ทดลองแบ่งกลุ่มของตัวอักษรเพื่อหาจำนวนและสมาชิกของกลุ่มที่เหมาะสม เพื่อให้ได้ค่าความถูกต้องเพิ่มขึ้นสำหรับการใช้วิธี HV
   - ทดลองเปลี่ยนค่าขอบเขตบน ขอบเขตล่าง และค่าขีดแบ่งให้มีความหลากหลายมากขึ้น

- หาค่าที่จะช่วยบ่งชี้ว่าผลลัพธ์ที่ได้จากการรวมแบบ AV มีความน่าจะเป็นที่จะผิดให้แม่นยำยิ่งขึ้น เพื่อที่จะสามารถกำหนดให้เน็ตเวิร์กเฉพาะทางทำงานได้เหมาะสมยิ่งขึ้น เช่น ค่าเฉลี่ยหรือค่าส่วนเบี่ยงเบนมาตรฐาน

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

รายการอ้างอิง

[1] Gonzalez, R.C. and Woods, R.E. Digital Image Processing. USA: Addison Wesley, 1993.

[2] ธเนศ ศรีวิรุฬห์ชัย. การรู้จำตัวอักษรพิมพ์ภาษาไทยโดยใช้เทคนิคด้านการวิเคราะห์ตัวประกอบสำคัญและนิวรอลเน็ตเวิร์ก. วิทยานิพนธ์ปริญญามหาบัณฑิต สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย, 2541.

[3] Limin Fu. Neural Networks in Computer Intelligence. Singapore: McGraw-Hill, 1994.

[4] Hansen, L., and Salamon, P. (1990). Neural Network Ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence Vol.12: 993-1001.

[5] Maclin, R., and Shavlik, J. (1995). Combining the Predictions of Multiple Classifiers: Using Competitive Learning to Initialize Neural Networks. Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence: 524-530. Montreal, Canada.

[6] Tumer, K., and Ghosh, J. (1996). Analysis of Decision Boundaries in Linearly combined Neural Classifiers. Pattern Recognition Vol.29 No.2:341-348.

[7] Breiman, L. (1996). Bagging Predictors. Machine Learning vol.24 No.2: 123-140.

[8] Parmanto, B., Munro, P. W., and Doyle, H. R. (1996). Improving Committee Diagnosis with Resampling Techniques. In Touretzky, D. S., Mozer, M. C.,and Hesselmo, M. E. (Eds.). Advance in Neural Information Processing systems (NIPS), Vol.8: 882-888 Cambridge, MA.: MIT Press.

[9] Freund, Y., and Schapire, R. E. (1995). A Decision-Theoretic Generalization of On-line Learning and an Application to Boosting. Technical report, AT&T Bell Laboratories, Murray Hill, NJ.

[10] Freund, Y., and Schapire, R. E. (1996). Experiments with a New Boosting Algorithm. In Saitta, L. (Ed.). Proceedings of the Thirteenth International Conference on Machine Learning (ICML-96): 148-156. San Francisco, CA.: Morgan Kaufmann.

[11] Freund, Y., and Schapire, R. E. (1999). A Short Introduction to Boosting. Journal of Japanese Society for Artificial Intelligence Vol.14 No.5: 771-780.

[12] Cherkauer, K. J. (1996). Human Expert-level Performance on a Scientific Image Analysis Task by a System Using Combined Artificial Neural Networks. In Chan, P. (Ed.). Working Notes of the AAAI Workshop on Integrating Multiple Learned Models: 15-21.

[13] Dietterich, T. G., and Bakiri, G. (1991). Error-Correcting Output Codes: A General Method for Improving Multiclass Inductive Learning Programs.

Proceedings of the 9th National Conference on Artificial Intelligence Vol.2: 572-577. Cambridge, Menlo Park: AAAI Press/MIT Press.

[14] Dietterich, T. G., and Bakiri, G. (1995). Solving Multiclass Learning Problems via Error-Correcting Output Codes. Journal of Artificial Intelligence Research Vol.2 No.2: 263-286.

[15] Kolen, J. F., and Pollack, J. B. (1991). Back Propagation is Sensitive to Initial Conditions. Advances in Neural Information Processing Systems Vol. 3: 860-867. San Francisco, CA.: Morgan Kaufmann.

[16] Maclin, R., and Opitz, D. (1997). An Empirical Evaluation of Bagging and Boosting. Proceedings of the Fourteenth National conference on Artificial Intelligence: 546-551. Cambridge/Menlo Park: AAAI Press/MIT Press.

[17] Dietterich, T. G. (1998). An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization. Machine Learning: 1-22.

[18] Opitz, D. W., and Shavlik, J. W. (1996). Actively Searching for an Effective Neural Network Ensemble. Connection Science Vol.8 No.3, 4: 337-353.

[19] Opitz, D. W., and Shavlik, J. W. (1996). Generating Accurate and Diverse Members of a Neural Network Ensemble. In Touretzky, D.S., Mozer, M. C.,and Hasselmo, M. E. (Eds.). Advances in Neural Information Processing Systems (NIPS) Vol.8: 535-541. Cambridge/Boston/London: MIT Press.

[20] Harries, M. (1999). Boosting a Strong Learner: Evidence Against the Minimum Margin. Proceedings of International Conference on Machine Learning (ICML-99): 171-180.

[21] Fan, W., Stolfo, S. J., Zhang, J. ,and Chan, P. K. (1999). AdaCost: Misclassification Cost-sensitive Boosting. Proceedings of International Conference on Machine Learning (ICML-99): 97-105.

[22] Clemen, R. T. (1989). Combining forcasts: A Review and Annotated Bibliography. International Journal of Forecasting Vol.5: 559-583.

[23] Jimenez, D., and Walsh, N. (1998). Dynamically Weighted Ensemble Neural Networks for Classification. Neural Networks Proceedings of the 1998 IEEE International Joint Conference on IEEEWorld Congress on Computational Intelligence Vol.1: 753-756.

[24] Jordan, M. L., and Jacobs, R. A. (1994). Hierarchical Mixtures of Experts and the EM Algorithm. Neural Computation Vol.6 No.2: 181-214.

[25] Wolpert, D. (1992). Stacked Generalization. Neural Networks Vol.5 No.2: 241-260.

[26] Dietterich, T. G. (1997). Machine Learning Research: Four Current Directions. AI Magazine Vol.18 No.4: 97-136.

ภาคผนวก

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ก

ตัวอักษรต้นแบบรูปแบบต่างๆ

ตัวอักษรต้นแบบรูปแบบ AngsanaUPC

ขนาด 16 จุด

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

ขนาด 18 จุด

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

ขนาด 20 จุด

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

ขนาด 22 จุด

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

ขนาด 24 จุด

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

ขนาด 26 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ฺ ็ ่ ้ ๊ ๋ เ ไ ใ ๐ ฤ ๅ ๆ ฯ ์ ๎ ๏ ๚ ๛

ขนาด 28 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ฺ ็ ่ ้ ๊ ๋ เ ไ ใ ๐ ฤ ๅ ๆ ฯ ์ ๎ ๏ ๚ ๛

ขนาด 36 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ
ฐ ฑ ฒ ณ ด ต ถ ท ธ น บ ป ผ ฝ พ
ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ ะ
า ิ ี ึ ื ุ ู ฺ ็ ่ ้ ๊ ๋ เ ไ ใ ๐
ฤ ๅ ๆ ฯ ์ ๎ ๏ ๚ ๛

ตัวอักษรต้นแบบรูปแบบ BrowaliaUPC

ขนาด 16 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ฺ ็ ั ์ เ โ ไ ใ ๆ ฯ ่ ้ ๊ ๋

ขนาด 18 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ฺ ็ ั ์ เ โ ไ ใ ๆ ฯ ่ ้ ๊ ๋

ขนาด 20 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ฺ ็ ั ์ เ โ ไ ใ ๆ ฯ ่ ้ ๊ ๋

ขนาด 22 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ฺ ็ ั ์ เ โ ไ ใ ๆ ฯ ่ ้ ๊ ๋

ขนาด 24 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ฺ ็ ั ์ เ โ ไ ใ ๆ ฯ ่ ้ ๊ ๋

ขนาด 26 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ่ ้ ๊ ๋ ์ ๎ เ โ ไ ใ ๐ ฤ ฦ ๆ ๚ ๛ ั ๑ ๗ +

ขนาด 28 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ่ ้ ๊ ๋ ์ ๎ เ โ ไ ใ ๐ ฤ ฦ ๆ ๚ ๛ ั ๑ ๗ +

ขนาด 36 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ
ฐ ฑ ฒ ณ ด ต ถ ท ธ น บ ป ผ ฝ พ
ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ ะ
า ิ ี ึ ื ุ ู ็ ่ ้ ๊ ๋ ์ ๎ เ โ ไ ใ ๐
ฤ ฦ ๆ ๚ ๛ ั ๑ ๗ +

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ตัวอักษรต้นแบบรูปแบบ CordiaUPC

ขนาด 16 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ

ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ

ะ า ิ ี ึ ื ุ ู ็ ั ์ เ แ ไ ใ ๆ ฤ ฦ ฯ ่ ้ ๊ ๋ +

ขนาด 18 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ

ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ

ะ า ิ ี ึ ื ุ ู ็ ั ์ เ แ ไ ใ ๆ ฤ ฦ ฯ ่ ้ ๊ ๋ +

ขนาด 20 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ

ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ

ะ า ิ ี ึ ื ุ ู ็ ั ์ เ แ ไ ใ ๆ ฤ ฦ ฯ ่ ้ ๊ ๋ +

ขนาด 22 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ

ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ

ะ า ิ ี ึ ื ุ ู ็ ั ์ เ แ ไ ใ ๆ ฤ ฦ ฯ ่ ้ ๊ ๋ +

ขนาด 24 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ

ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ

ะ า ิ ี ึ ื ุ ู ็ ั ์ เ แ ไ ใ ๆ ฤ ฦ ฯ ่ ้ ๊ ๋ +

ขนาด 26 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ

ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ

ะ า ิ ี ึ ื ุ ู ฺ ็ ั ์ เ โ ไ ใ ๆ ๆ ฯ ่ ้ ๊ ๋ +

ขนาด 28 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ

ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ

ะ า ิ ี ึ ื ุ ู ฺ ็ ั ์ เ โ ไ ใ ๆ ๆ ฯ ่ ้ ๊ ๋ +

ขนาด 36 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ

ฐ ฑ ฒ ณ ด ต ถ ท ธ น บ ป ผ ฝ พ

ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ ะ

า ิ ี ึ ื ุ ู ฺ ็ ั ์ เ โ ไ ใ

ๆ ๆ ฯ ่ ้ ๊ ๋ +

ตัวอักษรต้นแบบรูปแบบ DilleniaUPC

ขนาด 16 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ ไ ใ ๆ ฤ ฦ ๆ ฯ ่ ้ ๊ ๋

ขนาด 18 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ ไ ใ ๆ ฤ ฦ ๆ ฯ ่ ้ ๊ ๋

ขนาด 20 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ ไ ใ ๆ ฤ ฦ ๆ ฯ ่ ้ ๊ ๋

ขนาด 22 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ ไ ใ ๆ ฤ ฦ ๆ ฯ ่ ้ ๊ ๋

ขนาด 24 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ ไ ใ ๆ ฤ ฦ ๆ ฯ ่ ้ ๊ ๋

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ขนาด 26 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ

ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ

ะ า ิ ี ึ ื ุ ู ฺ เ แ โ ใ ไ ๅ ๆ ็ ่ ้ ๊ ๋ ์ ๎ ๏ ๚ ๛

ขนาด 28 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ

ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ

ะ า ิ ี ึ ื ุ ู ฺ เ แ โ ใ ไ ๅ ๆ ็ ่ ้ ๊ ๋ ์ ๎ ๏ ๚ ๛

ขนาด 36 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ

ฐ ฑ ฒ ณ ด ต ถ ท ธ น บ ป ผ ฝ พ

ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ ะ

า ิ ี ึ ื ุ ู ฺ เ แ โ ใ ไ ๅ

ๆ ็ ่ ้ ๊ ๋ ์ ๎ ๏ ๚ ๛

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ตัวอักษรต้นแบบรูปแบบ EucrosiaUPC

ขนาด 16 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ โ ไ ใ ๆ ๆ ๆ ่ ้ ๊ ๋ +

ขนาด 18 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ โ ไ ใ ๆ ๆ ๆ ่ ้ ๊ ๋ +

ขนาด 20 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ โ ไ ใ ๆ ๆ ๆ ่ ้ ๊ ๋ +

ขนาด 22 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ โ ไ ใ ๆ ๆ ๆ ่ ้ ๊ ๋ +

ขนาด 24 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ โ ไ ใ ๆ ๆ ๆ ่ ้ ๊ ๋ +

ขนาด 26 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ่ ้ ๊ ๋ เ โ ไ ใ ๆ ๆ ๆ ๆ ๆ ๆ

ขนาด 28 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ่ ้ ๊ ๋ เ โ ไ ใ ๆ ๆ ๆ ๆ ๆ ๆ

ขนาด 36 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ
ฐ ฑ ฒ ณ ด ต ถ ท ธ น บ ป ผ ฝ พ
ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ ะ
า ิ ี ึ ื ุ ู ็ ่ ้ ๊ ๋ เ โ ไ ใ
ๆ ๆ ๆ ๆ ๆ ๆ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ตัวอักษรต้นแบบรูปแบบ FreesiaUPC

ขนาด 16 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ โ ไ ใ ๆ ฤ ฦ ๆ ๆ ่ ้ ๊ ๋ +

ขนาด 18 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ โ ไ ใ ๆ ฤ ฦ ๆ ๆ ่ ้ ๊ ๋ +

ขนาด 20 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ โ ไ ใ ๆ ฤ ฦ ๆ ๆ ่ ้ ๊ ๋ +

ขนาด 22 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ โ ไ ใ ๆ ฤ ฦ ๆ ๆ ่ ้ ๊ ๋ +

ขนาด 24 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ โ ไ ใ ๆ ฤ ฦ ๆ ๆ ่ ้ ๊ ๋ +

ขนาด 26 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ

ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ แ โ ไ ใ ํ ฤ ฦ ๆ ฯ ่ ้ ๊ ๋

ขนาด 28 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ ฒ ณ ด ต ถ
ท ธ น บ ป ผ ฝ พ ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ
ะ า ิ ี ึ ื ุ ู ็ ั ์ เ แ โ ไ ใ ํ ฤ ฦ ๆ ฯ ่ ้ ๊ ๋

ขนาด 36 จุด

ก ข ฃ ค ฅ ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ
ฐ ฑ ฒ ณ ด ต ถ ท ธ น บ ป ผ ฝ พ
ฟ ภ ม ย ร ล ว ศ ษ ส ห ฬ อ ฮ ะ
า ิ ี ึ ื ุ ู ็ ั ์ เ แ โ ไ ใ ํ
ฤ ฦ ๆ ฯ ่ ้ ๊ ๋

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาคผนวก ข

ตารางแสดงผลของตัวอักษรที่ผิดแบ่งตามรูปแบบและขนาด

ตารางที่ ข.1 แสดงความผิดพลาดแยกตามรูปแบบและขนาดที่ได้จากการรวมผลลัพธ์ด้วยวิธีต่างๆ เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 1*

รูปแบบตัวอักษร	**ขนาด**	**จำนวน**	MV	MVP	AV	MMWV	CWV	HV12	HV9
AngsanaUPC	36	68	2	3	3	3	3	3	3
	28	68	6	6	6	6	6	6	7
	26	68	3	2	2	2	2	2	2
	24	68	2	1	1	1	1	2	2
	22	68	1	1	1	1	1	0	2
	20	68	1	1	1	2	1	1	1
	18	68	0	0	0	0	0	0	0
	16	68	1	1	1	1	1	1	1
รวม		544	16	15	15	16	15	15	18
BrowaliaUPC	36	68	2	3	2	2	2	2	2
	28	68	0	0	0	0	0	0	0
	26	68	0	0	0	0	1	0	0
	24	68	3	3	4	3	4	4	4
	22	68	1	2	2	2	2	2	2
	20	68	0	0	0	0	0	0	0
	18	68	0	0	0	0	0	0	0
	16	68	3	3	4	3	4	2	3
รวม		544	9	11	12	10	13	10	11
CordiaUPC	36	68	3	3	3	4	2	1	4
	28	68	7	7	7	8	7	6	7
	26	68	1	0	0	0	0	0	0
	24	68	3	3	3	3	2	2	3
	22	68	2	2	2	2	1	1	2
	20	68	3	3	3	3	3	1	2
	18	68	2	1	1	1	1	1	1
	16	68	3	0	0	0	0	0	0
รวม		544	24	19	19	21	16	12	19

ตารางที่ ข.1 แสดงความผิดพลาดแยกตามรูปแบบและขนาดที่ได้จากการรวมผลลัพธ์ด้วยวิธีต่างๆ เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 1* (ต่อ)

รูปแบบตัวอักษร	**ขนาด**	**จำนวน**	MV	MVP	AV	MMWV	CWV	HV12	HV9
DilleniaUPC	36	68	5	3	3	2	3	3	3
	28	68	6	5	5	5	5	5	5
	26	68	4	5	5	5	5	5	4
	24	68	7	4	4	4	4	4	3
	22	68	5	4	3	4	3	4	2
	20	68	0	0	0	0	0	0	1
	18	68	4	3	1	1	1	0	2
	16	68	2	3	3	3	3	3	3
รวม		544	33	27	24	24	24	24	23
EucrosiaUPC	36	68	3	3	3	3	3	2	2
	28	68	0	1	0	0	0	1	1
	26	68	2	2	2	1	1	2	1
	24	68	0	0	0	0	0	0	1
	22	68	0	0	0	0	0	0	0
	20	68	0	0	0	0	0	0	0
	18	68	1	0	0	0	0	0	0
	16	68	2	1	1	1	1	0	1
รวม		544	8	7	6	5	5	5	6
FreesiaUPC	36	68	2	3	2	2	2	2	2
	28	68	0	0	0	0	0	0	0
	26	68	1	1	1	1	1	0	0
	24	68	3	1	1	1	1	0	0
	22	68	0	0	0	0	0	0	0
	20	68	2	2	1	1	1	0	1
	18	68	0	0	0	0	0	0	0
	16	68	1	1	1	1	1	1	1
รวม		544	9	8	6	6	6	3	4
รวมทุกรูปแบบ		**3264**	**99**	**87**	**82**	**82**	**79**	**69**	**81**

ตารางที่ ข.2 แสดงความผิดพลาดแยกตามรูปแบบและขนาดที่ได้จากการรวมผลลัพธ์ด้วยวิธีต่างๆ เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 2*

รูปแบบตัวอักษร	**ขนาด**	**จำนวน**	MV	MVP	AV	MMWV	CWV	HV12	HV9
AngsanaUPC	36	68	8	5	4	4	4	4	4
	28	68	5	5	5	5	5	5	5
	26	68	2	1	1	1	1	0	1
	24	68	2	1	1	1	1	1	1
	22	68	3	2	1	1	1	1	0
	20	68	1	1	0	0	0	0	0
	18	68	0	0	0	0	0	0	0
	16	68	1	0	0	0	0	0	0
รวม		544	22	15	12	12	12	11	11
BrowaliaUPC	36	68	10	9	7	8	6	7	7
	28	68	1	2	1	1	1	0	0
	26	68	0	0	0	0	0	0	0
	24	68	0	0	0	0	0	0	0
	22	68	0	0	0	0	0	0	0
	20	68	0	0	0	0	0	0	0
	18	68	0	0	0	0	0	0	0
	16	68	1	0	0	0	1	0	0
รวม		544	12	11	8	9	8	7	7
CordiaUPC	36	68	9	7	7	7	7	4	6
	28	68	4	4	3	3	3	4	3
	26	68	1	0	0	1	0	0	0
	24	68	1	1	1	0	0	1	1
	22	68	1	1	1	1	1	1	1
	20	68	1	3	3	3	3	2	2
	18	68	1	1	1	1	1	1	1
	16	68	4	3	3	3	3	2	3
รวม		544	22	20	19	19	18	15	17

ตารางที่ ข.2 แสดงความผิดพลาดแยกตามรูปแบบและขนาดที่ได้จากการรวมผลลัพธ์ด้วยวิธีต่างๆ เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 2* (ต่อ)

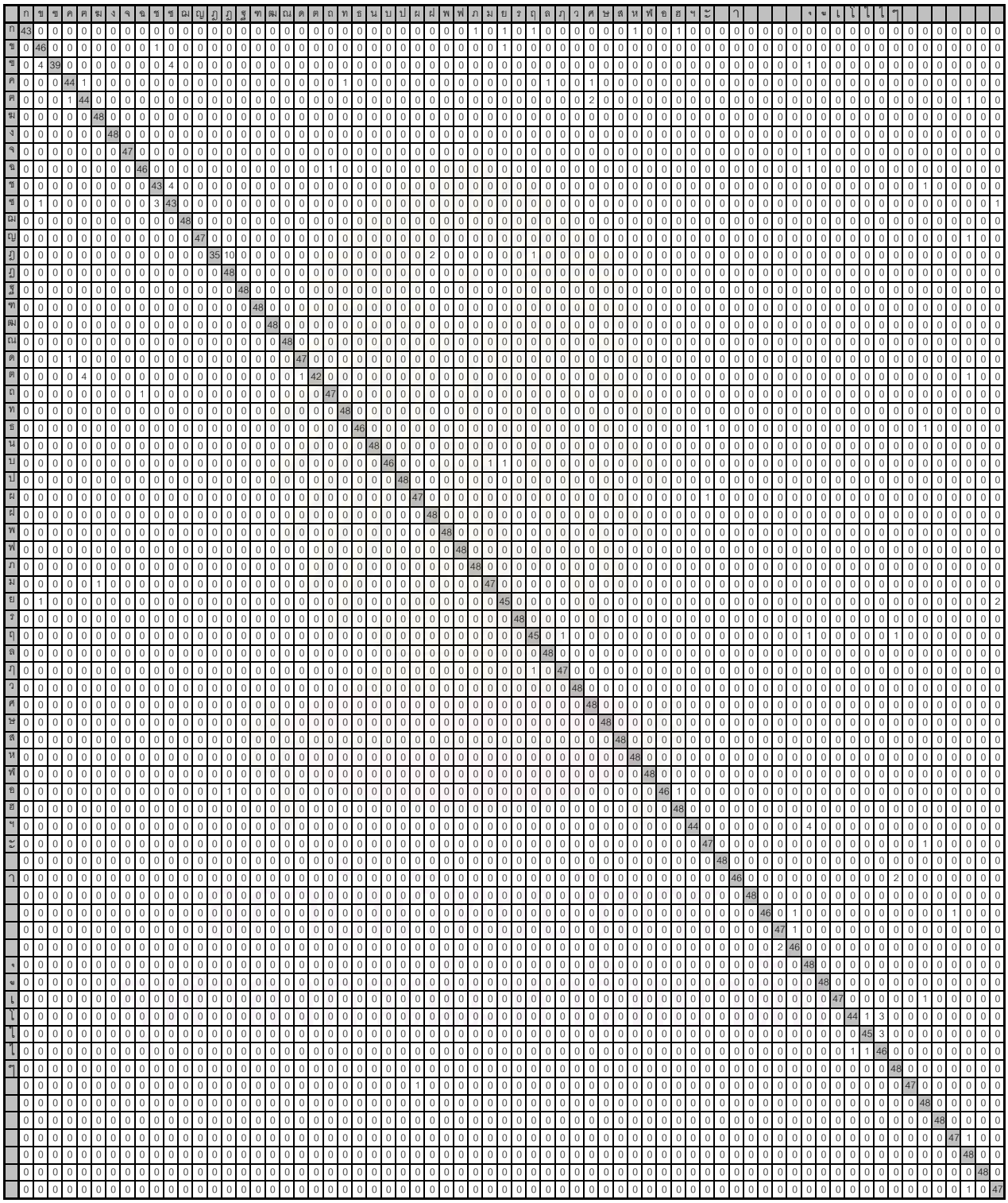
รูปแบบตัวอักษร	**ขนาด**	**จำนวน**	MV	MVP	AV	MMWV	CWV	HV12	HV9
DilleniaUPC	36	68	5	6	3	4	3	3	2
	28	68	2	2	2	2	2	2	1
	26	68	0	0	1	1	0	1	0
	24	68	3	2	1	1	1	1	3
	22	68	0	0	0	0	0	0	0
	20	68	0	0	0	0	0	1	1
	18	68	0	0	0	0	0	0	0
	16	68	4	3	4	4	3	1	2
รวม		544	14	13	11	12	9	9	9
EucrosiaUPC	36	68	6	4	4	5	4	4	4
	28	68	1	1	1	1	1	1	1
	26	68	2	1	1	1	1	0	0
	24	68	0	0	0	0	0	0	0
	22	68	0	0	0	0	0	0	0
	20	68	1	1	1	0	1	1	1
	18	68	0	0	0	0	0	0	0
	16	68	1	1	1	1	1	1	1
รวม		544	11	8	8	8	8	7	7
FreesiaUPC	36	68	6	7	7	7	7	6	7
	28	68	4	3	3	3	3	3	3
	26	68	4	1	1	1	2	0	0
	24	68	1	1	2	2	2	1	2
	22	68	0	0	0	0	0	0	0
	20	68	2	2	2	2	2	2	2
	18	68	1	1	1	1	1	1	1
	16	68	0	0	0	0	0	0	0
รวม		544	18	15	16	16	17	13	15
รวมทุกรูปแบบ		**3264**	**99**	**82**	**74**	**76**	**72**	**62**	**66**

ภาคผนวก ค

ตารางแสดงผลของตัวอักษรที่ผิดแบ่งตามตัวอักษร

ตารางที่ ค.1 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี MV เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 1*

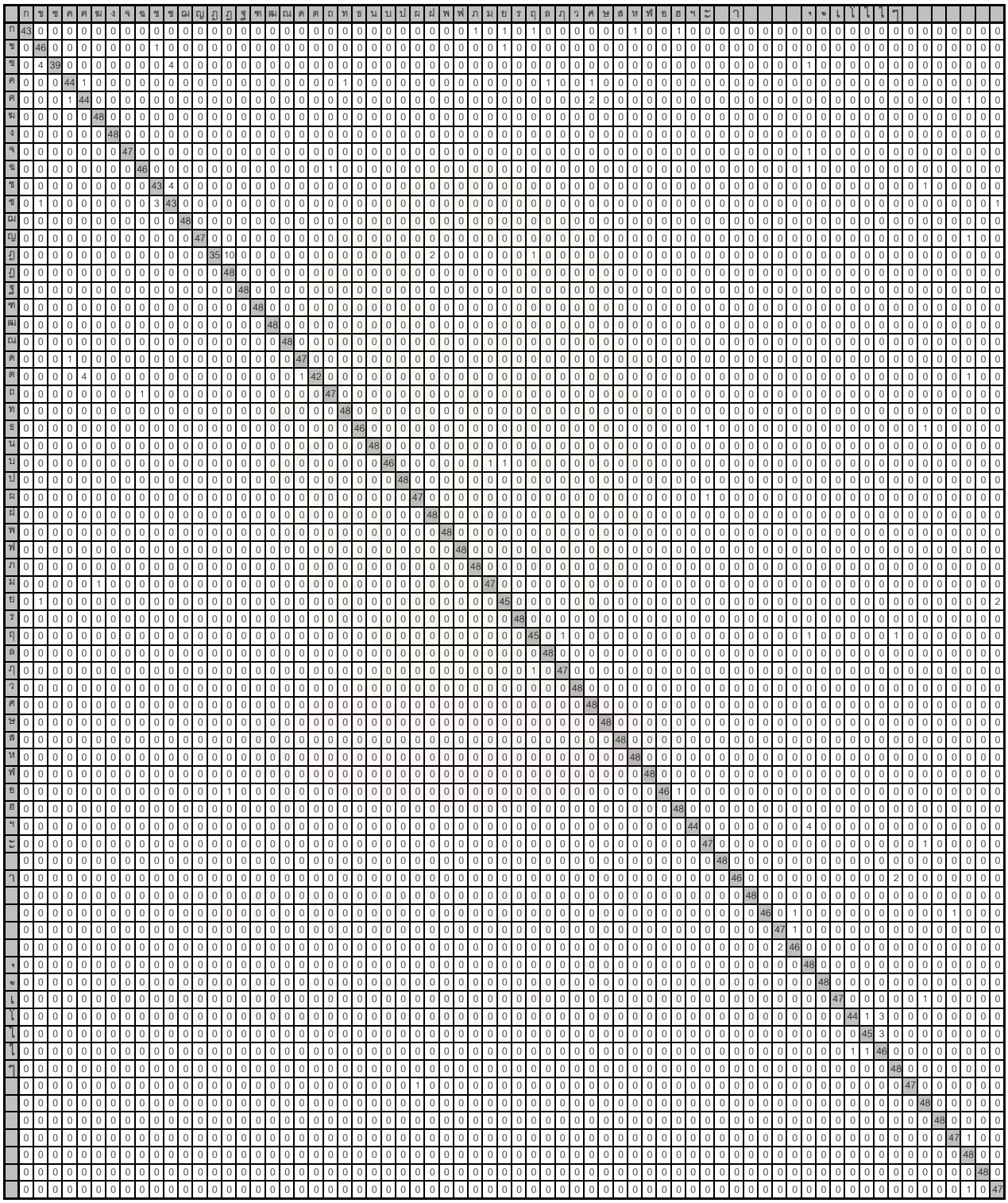
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.2 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี MV เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 2*

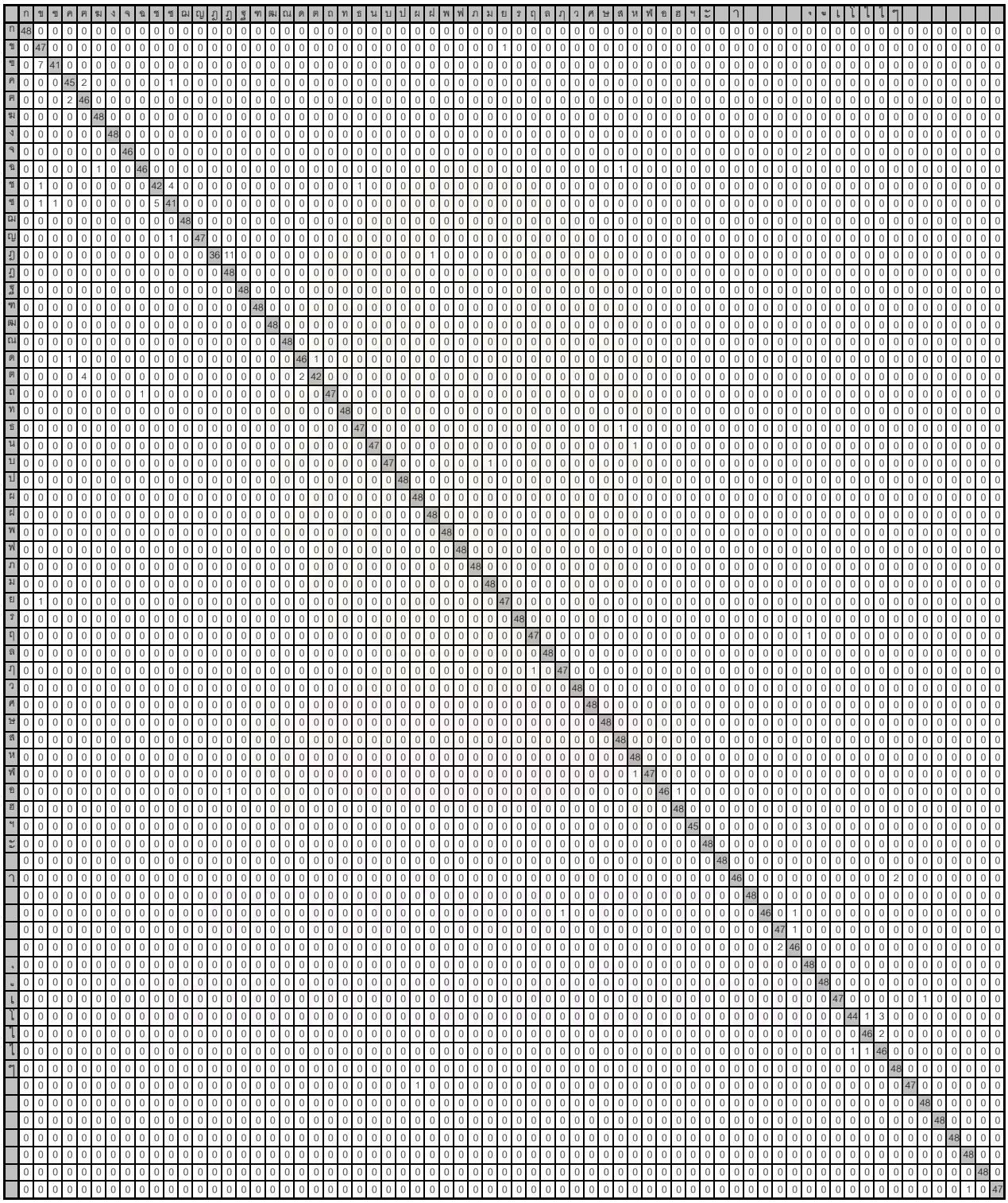
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.3 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี MVP เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 1*

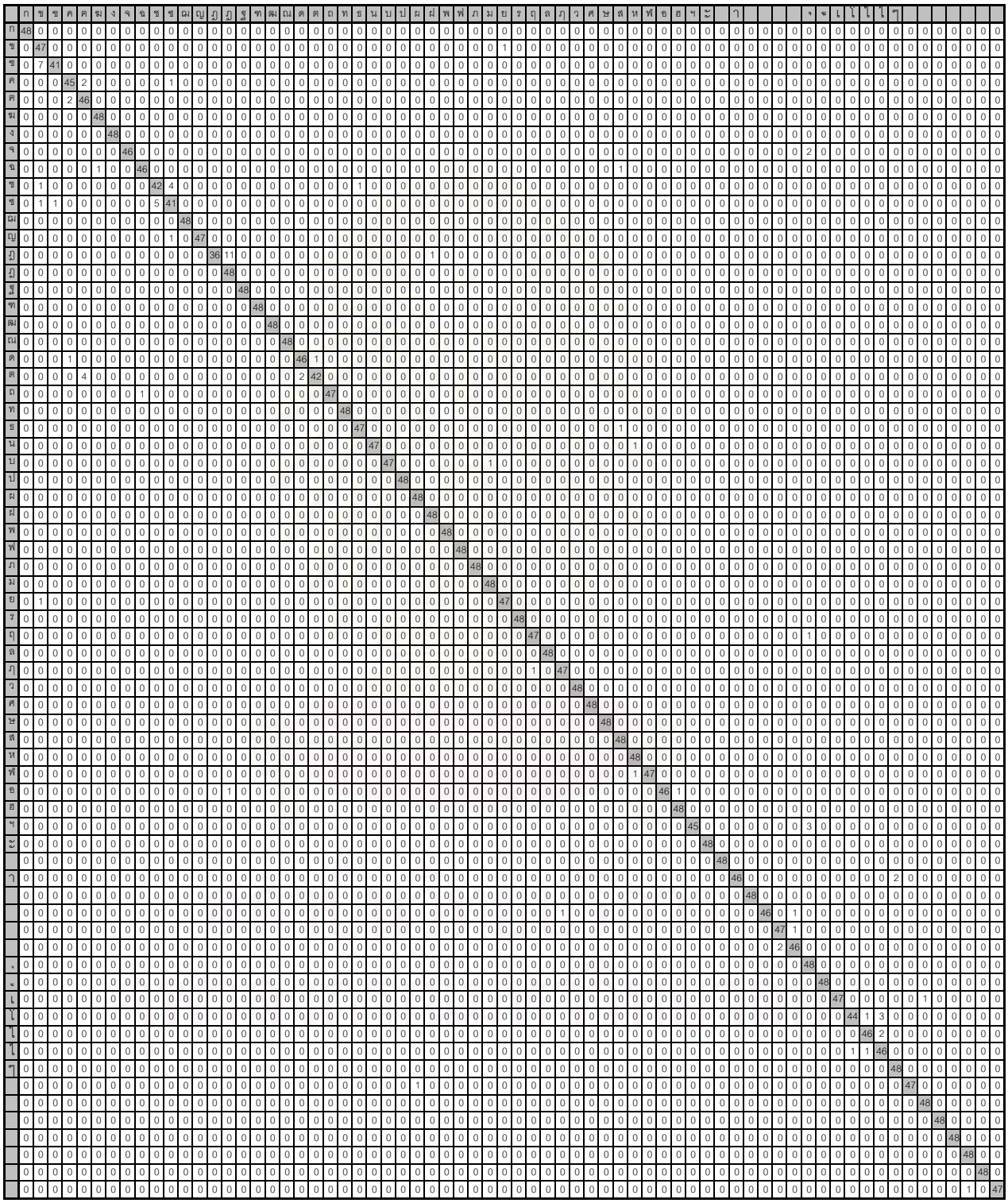
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.4 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี MVP เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 2*

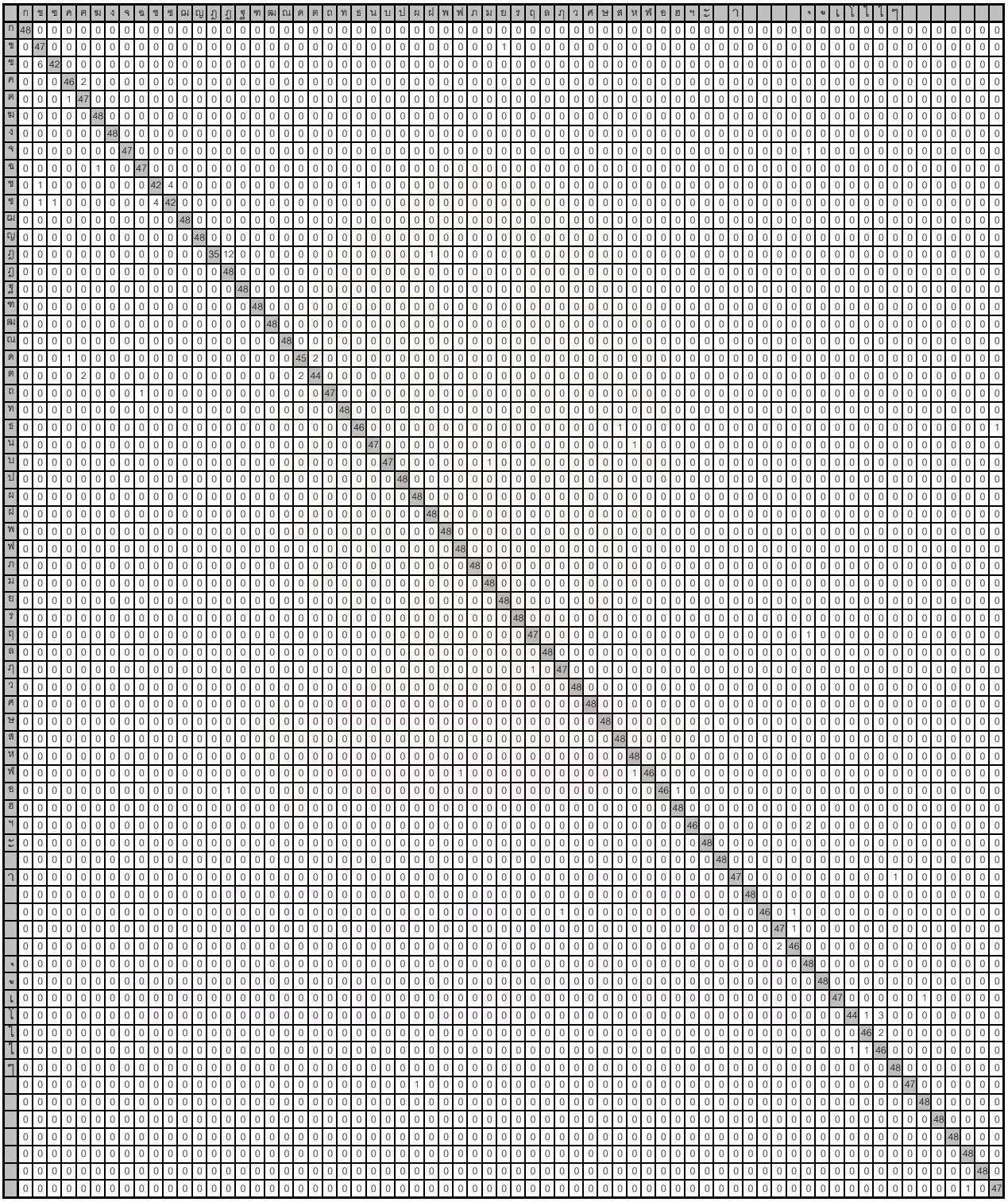
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.5 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี AV เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 1*

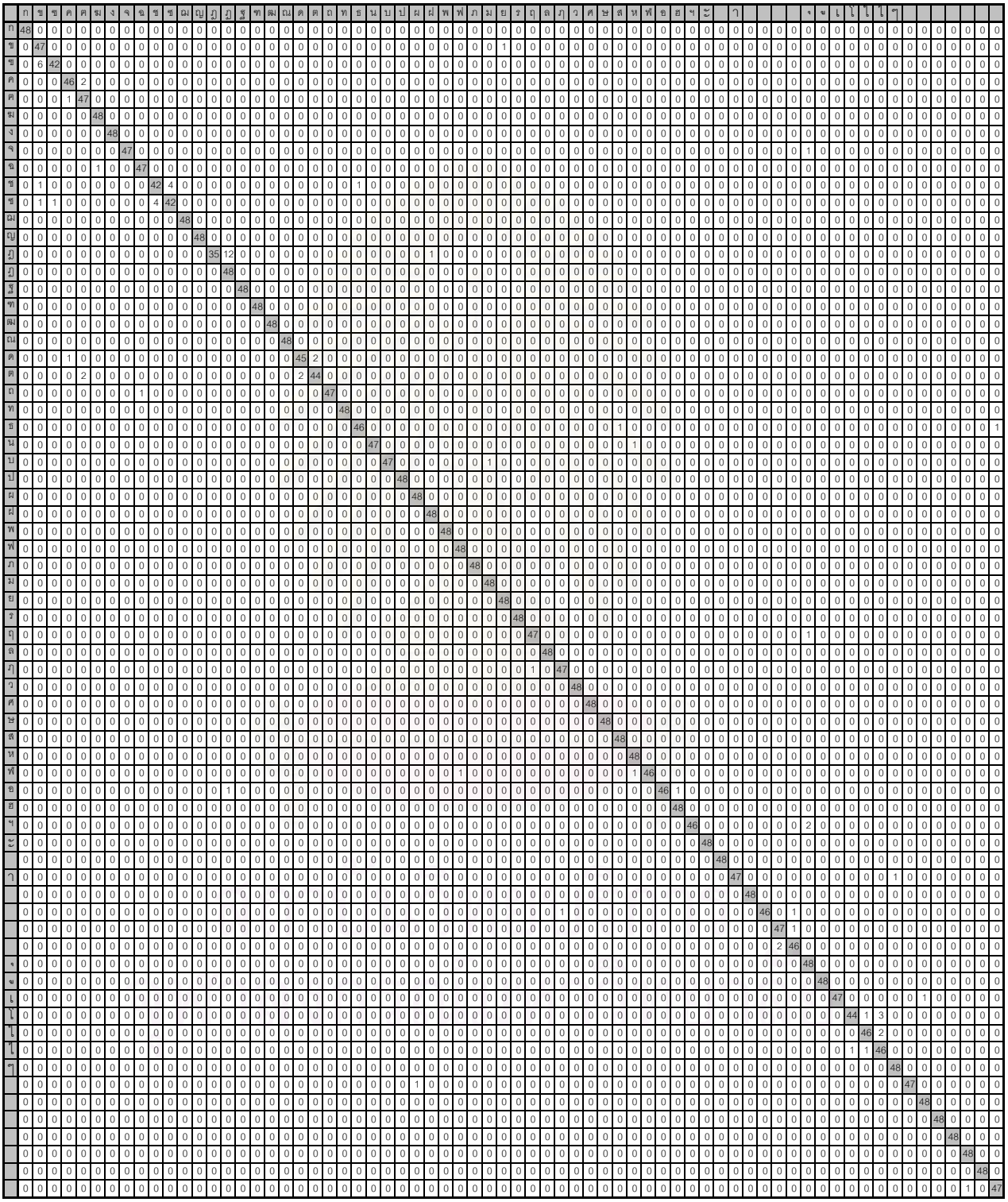
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.6 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี AV เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 2*

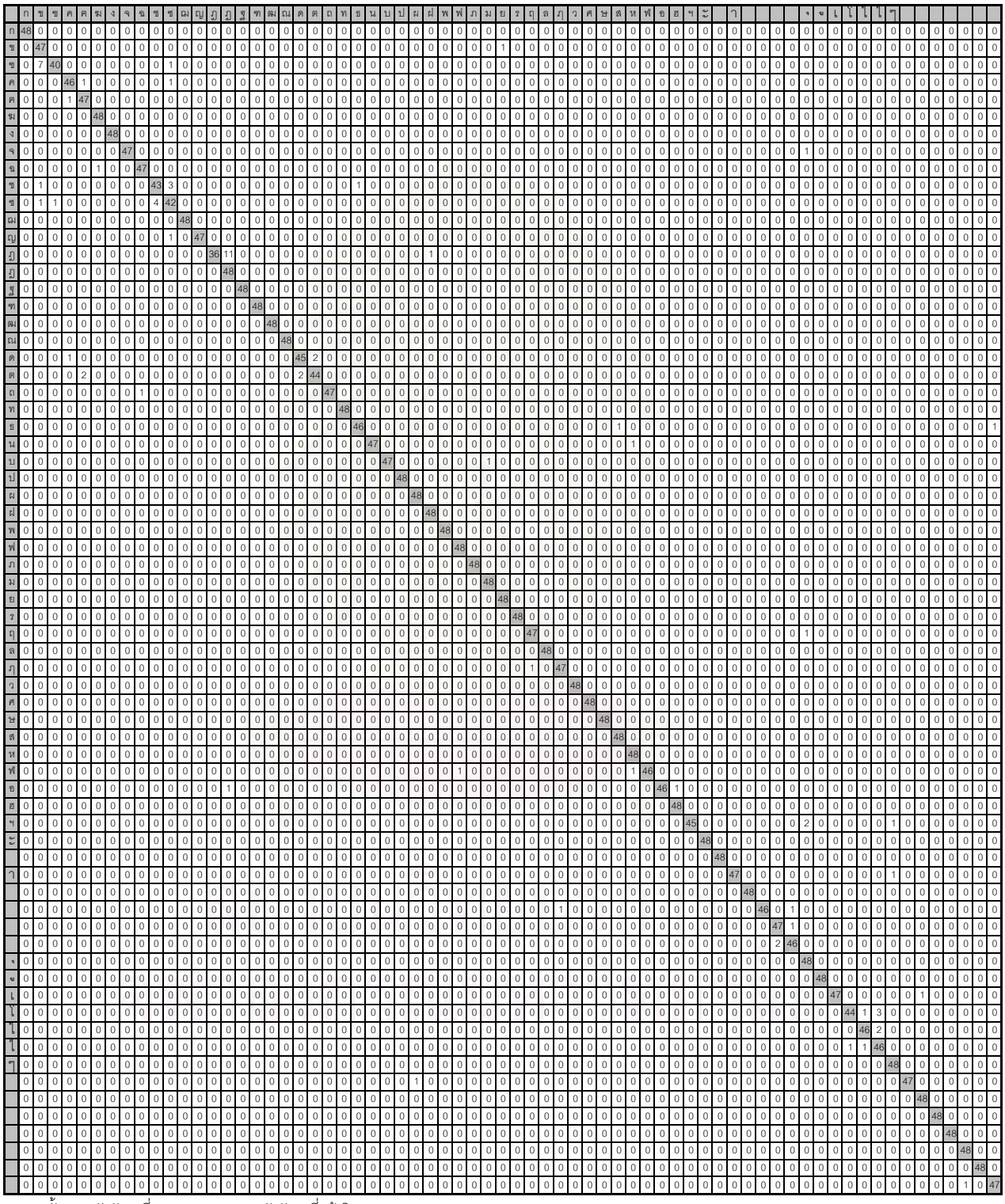
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.7 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี MMWV เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 1*

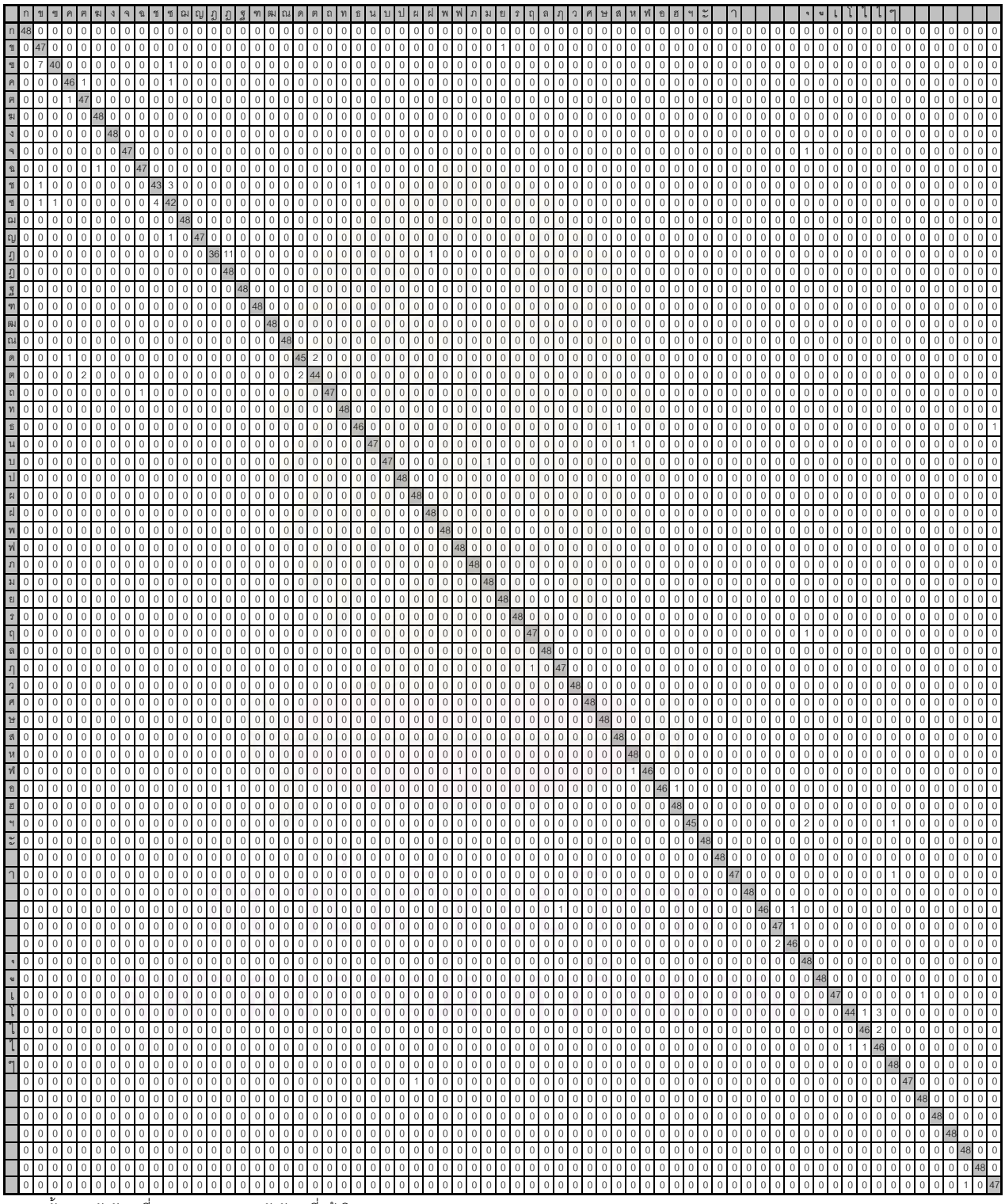
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.8 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี MMWV เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 2*

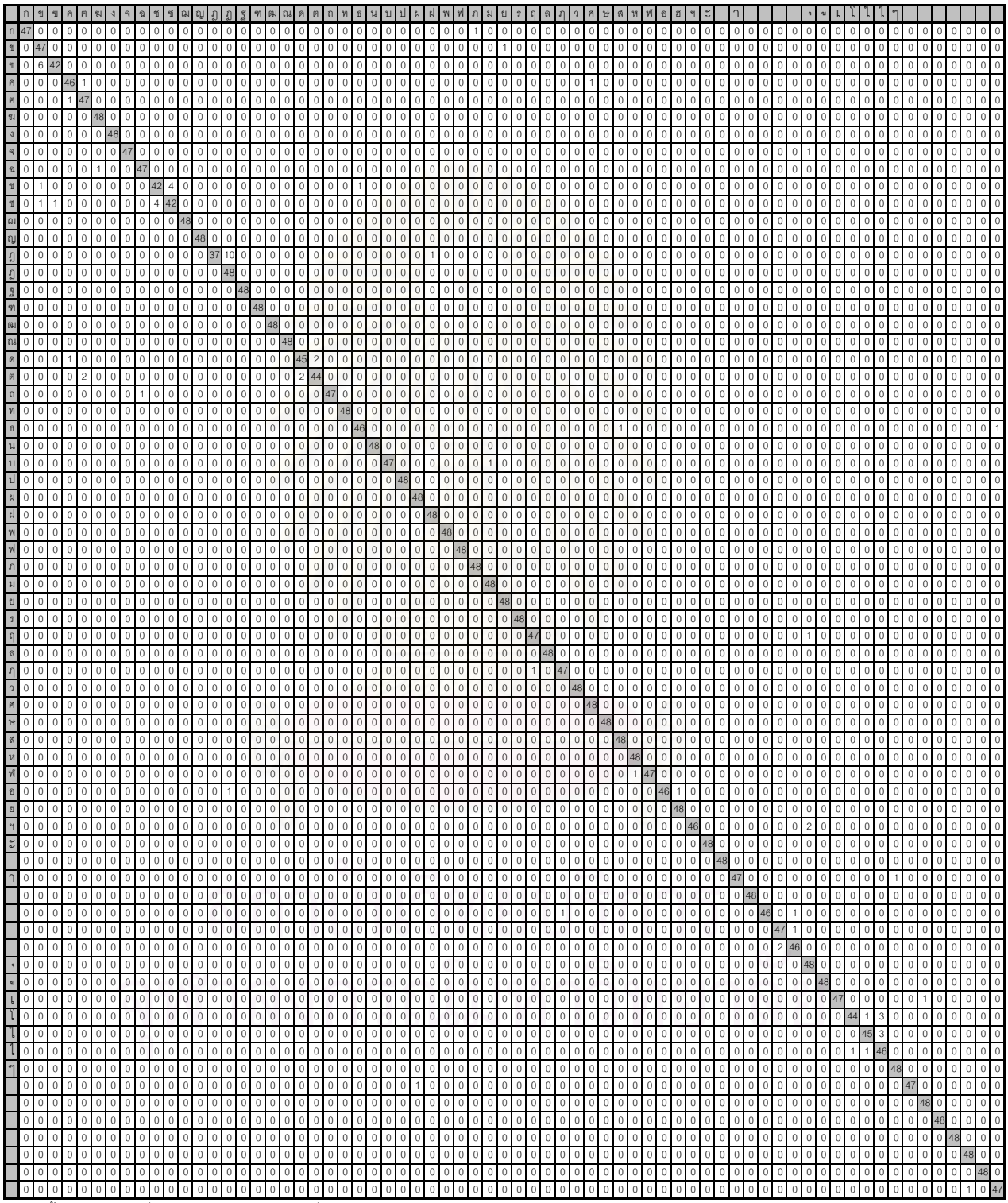
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.9 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี CWV เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 1*

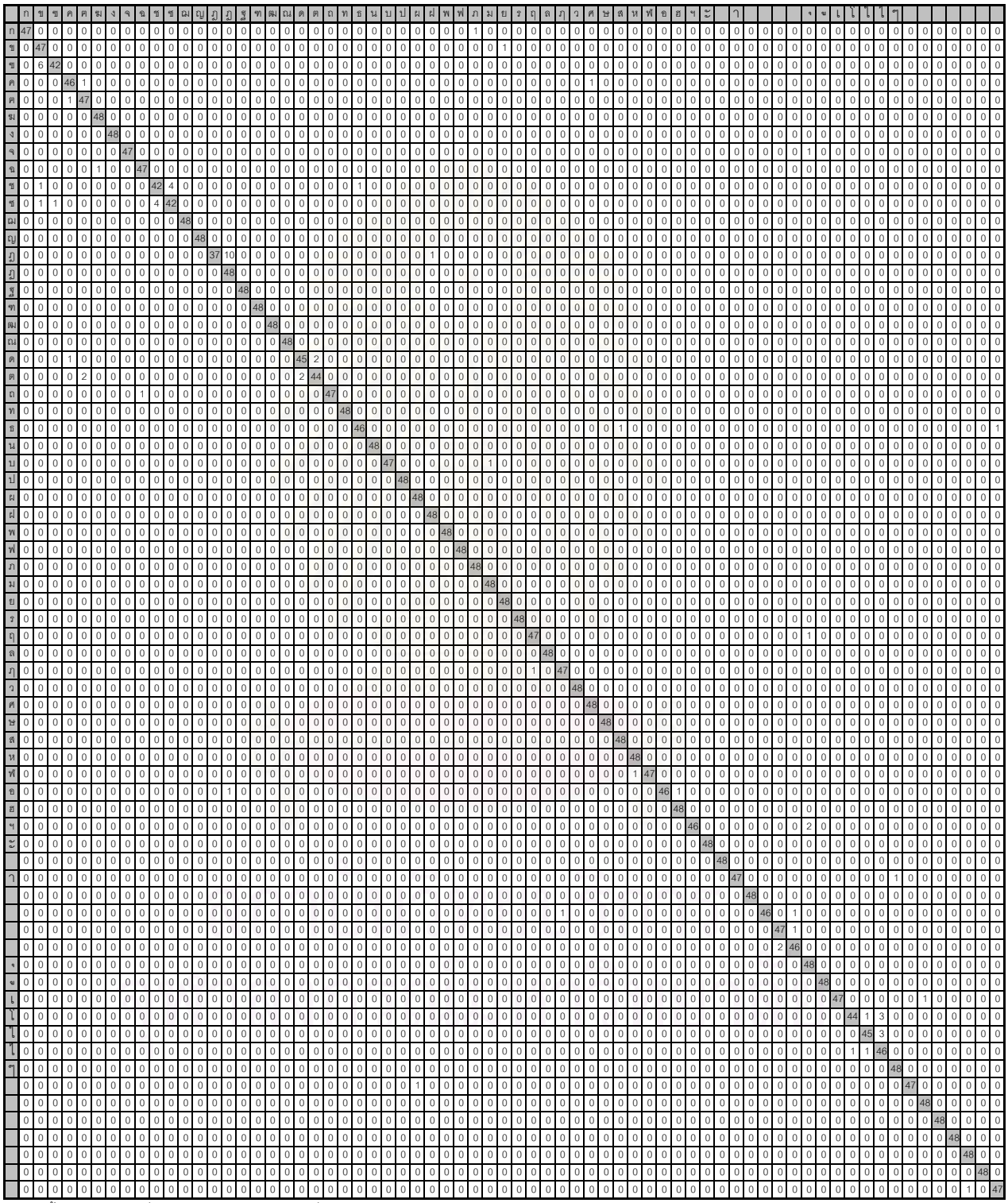
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.10 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี CWV เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 2*

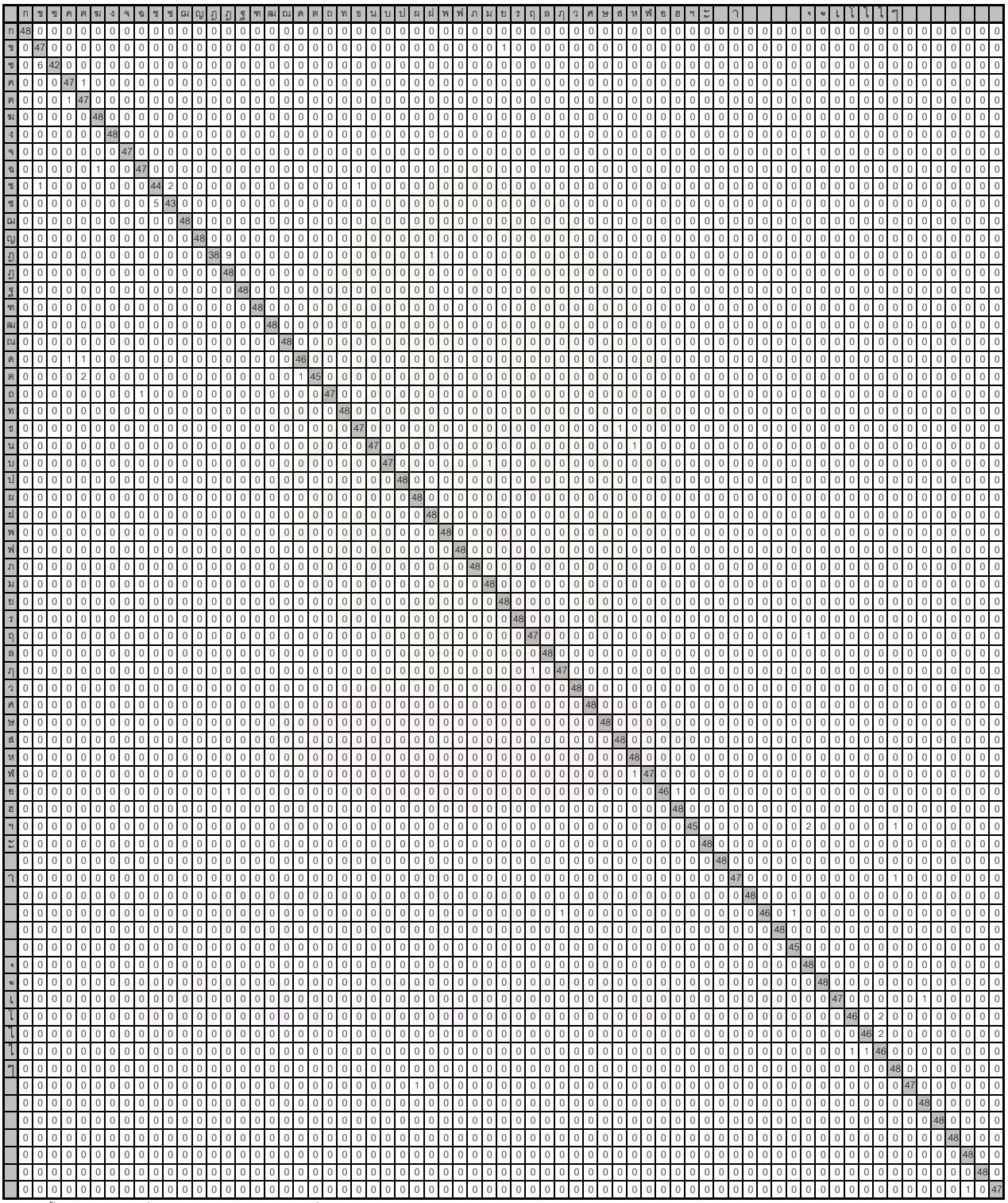
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.11 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี HV(12 กลุ่ม) เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 1*

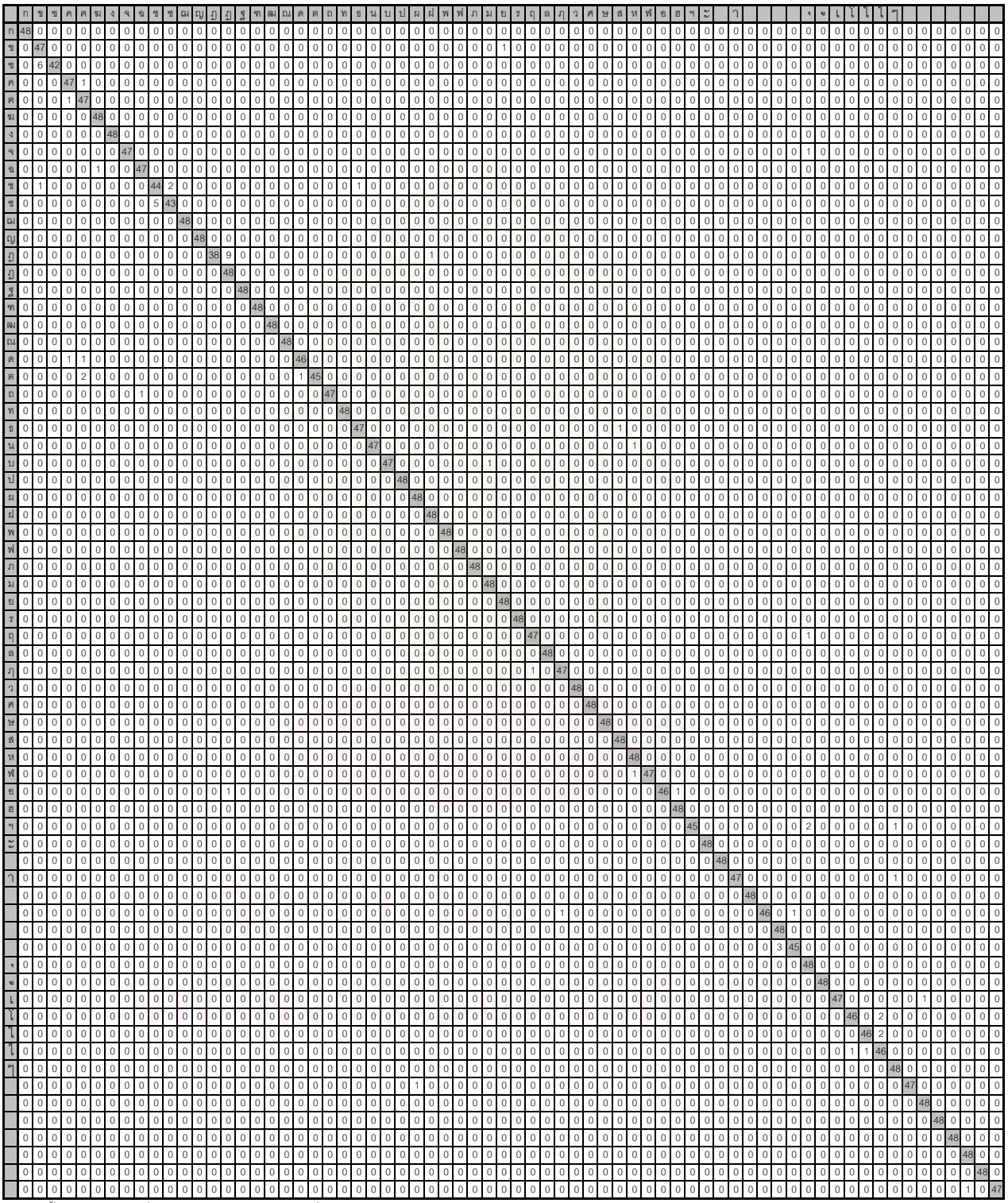
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.12 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี HV(12 กลุ่ม) เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 2*

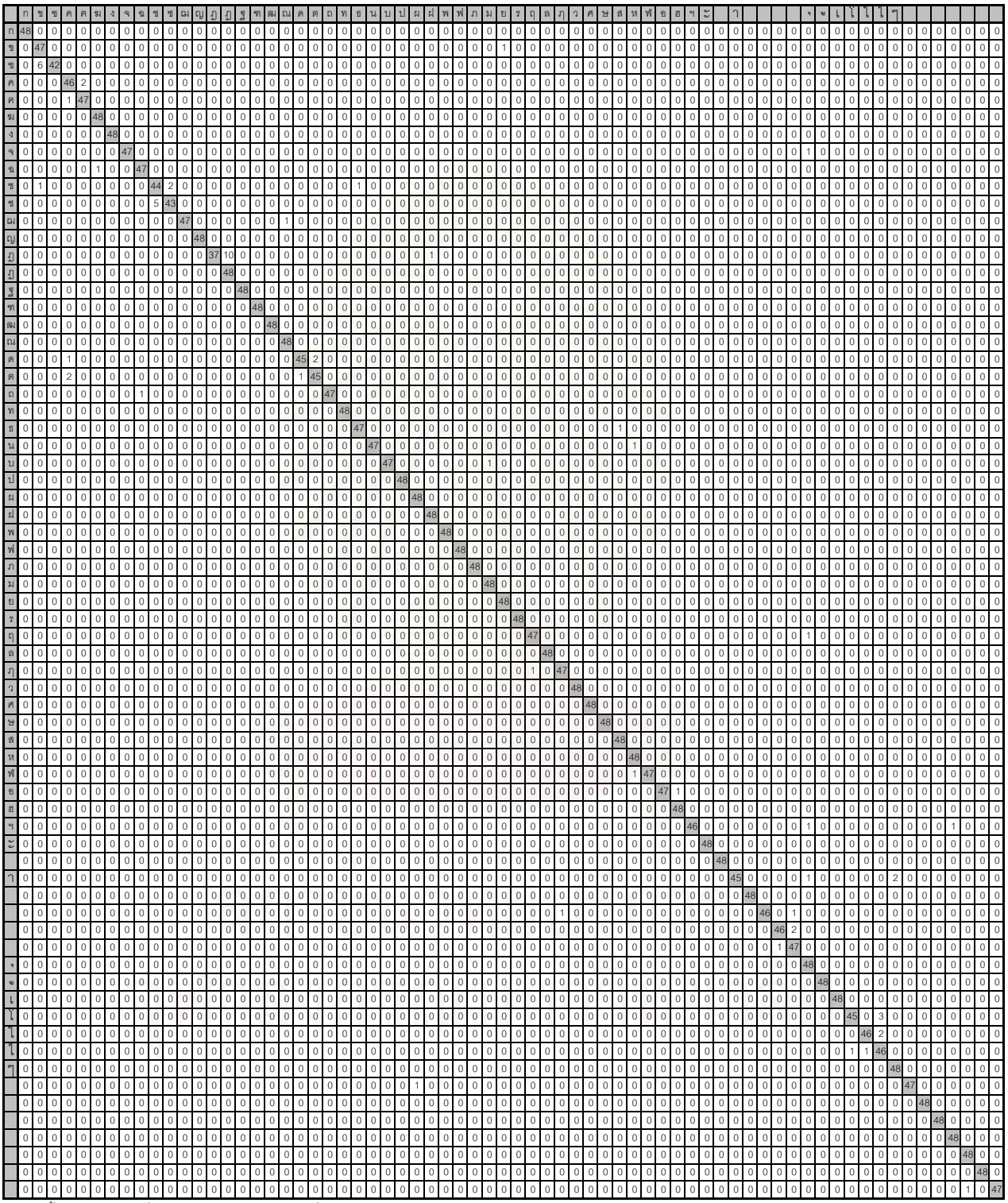
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.13 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี HV(9 กลุ่ม) เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 1*

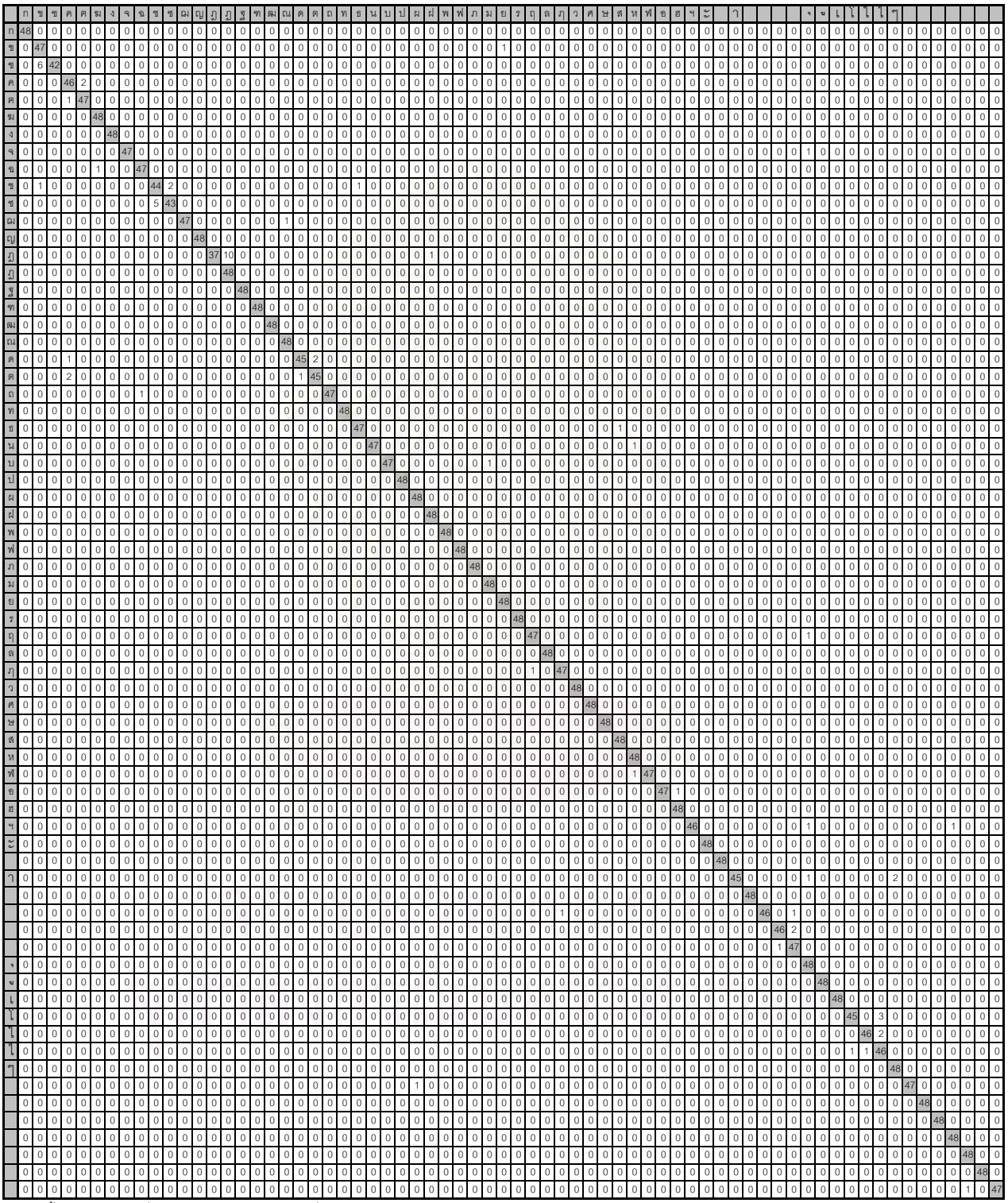
สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ตารางที่ ค.14 แสดงความผิดพลาดแบ่งตามตัวอักษรที่ได้จากการรวมผลลัพธ์ด้วยวิธี HV(9 กลุ่ม) เมื่อใช้ 3 เน็ตเวิร์ก บน *ชุดทดสอบที่ 2*

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



* แนวตั้งแสดงตัวอักษรที่ตอบ แนวนอนแสดงตัวอักษรที่แท้จริง

ประวัติผู้เขียนวิทยานิพนธ์

นางสาวสุขวสา พิชิตเดช เกิดเมื่อวันที่ 6 สิงหาคม พ.ศ. 2521 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ จากภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2541 และศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ ที่ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ปีการศึกษา 2542

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย