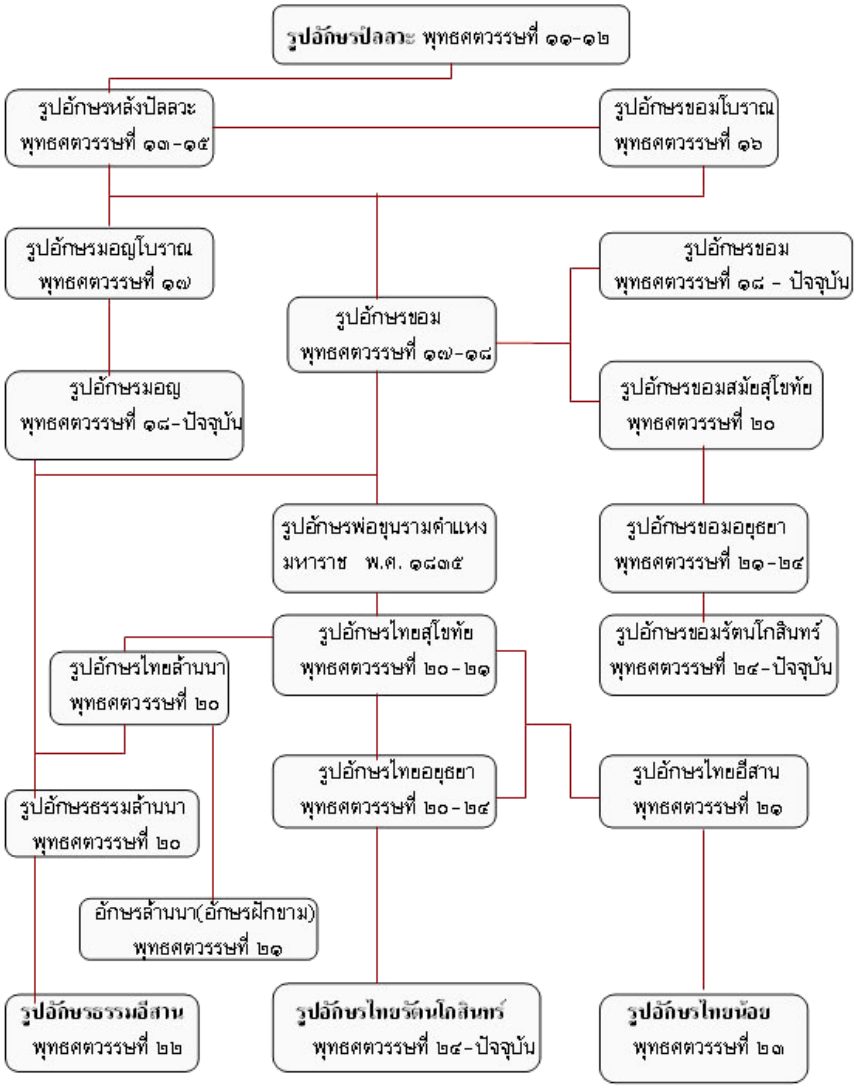
บทที่ 2

วรรณกรรมและงานวิจัยที่เกี่ยวข้อง

2.1 ความเป็นมาของอักษรธรรมอีสาน

ตัวอักษรธรรมอีสานเป็นตัวอักษรที่มีลักษณะคล้ายตัวอักษรล้านนาทางภาคเหนือ จะแตกต่างกันเป็นบางตัวเท่านั้น ซึ่งมีพัฒนาการมาจากอักษรมอญโบราณ โดยศาสตราจารย์ธวัช ปุณโณทก [3] กล่าวไว้ว่า อักษรธรรมอีสานและอักษรล้านนาหรืออักษรยวนนั้นได้สืบทอดมาจากอักษรมอญโบราณที่หริภุญไชยราวพุทธศตวรรษที่ 18



ภาพที่ 2.1 แสดงการกำเนิดตัวอักษรตามแนวของ ก่องแก้ว วีระประจักษ์ [1]

---

[1] http://www.esansawang.in.th/tham/thamhome.htm

จากภาพที่ 2.1 จะเห็นการกำเนิดตัวอักษรอีกแนวหนึ่ง เห็นที่มาของอักษรธรรมและอักษรไทยน้อยได้อย่างชัดเจน

มีการพบหลักฐานการใช้อักษรธรรมอีสานเก่าแก่ที่สุดในพุทธศตวรรษที่ 22 เป็นอักษรร่วมสมัยอยุธยา คือ จารึกวัดศรีคุณเมือง อำเภอเมือง จังหวัดหนองคาย เป็นศิลาทรายแดงสลักเป็นรูปใบเสมา สร้างเมื่อ พ.ศ. 2103 ส่วนหลักฐานที่เป็นใบลานนั้น เชื่อว่า คัมภีร์วิสุทธิมรรค ซึ่งจารเป็นภาษาบาลีล้วน เมื่อ พ.ศ. 2143 ตรงกับสมัยสมเด็จพระนเรศวรมหาราช จัดเป็นเอกสารที่เก่าแก่ที่สุดพบที่จังหวัดหนองคาย ปัจจุบันเก็บไว้ที่หอสมุดแห่งชาติ (เพ็ญพักตร์ ลิ้มสัมพันธ์. 2527:1) ส่วนหลักฐานที่พบที่ประเทศสาธารณรัฐประชาธิปไตยประชาชนลาวที่เก่าแก่ที่สุด ได้แก่จารึกบนฐานพระพุทธรูปวัดสีสะเกด ในนครเวียงจันทร์ (พ.ศ. 2033) ตรงกับสมัยพระราชแสนไทย รองลงมาได้แก่จารึกบนฐานพระพุทธรูปวัดพันหลวง ที่หลวงพระบาง (พ.ศ. 2067) ตรงกับสมัยพระเจ้าโพธิสารราช และสาเหตุที่ชื่อว่าตัวอักษรธรรมนั้นก็เพราะว่าตัวอักษรนี้ใช้เขียนบันทึกเรื่องราวเกี่ยวกับพระพุทธศาสนา เช่น พระไตรปิฎก พระธรรมคัมภีร์ต่างๆ เป็นต้น เพราะถือว่าเป็นอักษรชั้นสูง และมีความศักดิ์สิทธิ์

2.2 รูปแบบของตัวอักษรธรรมอีสาน [4]

อักษรธรรมอีสานมีลักษณะทั่วไปคล้ายกับอักษรล้านนา แต่อักขรวิธีอักษรธรรมอีสานเป็นแบบผสมระหว่างอักขรวิธีล้านนาผสมกับอักขรวิธีอักษรไทน้อย ซึ่งรูปลักษณะของอักษรธรรมอีสานเท่าที่พบมีดังนี้

2.2.1 รูปพยัญชนะ ประกอบด้วย

1) *รูปพยัญชนะเต็ม* คือ พยัญชนะที่เขียนเต็มตามรูปแบบปกติ มี 37 ตัว ทำหน้าที่เป็นพยัญชนะต้น โดยจะแบ่งตามวรรคแบบภาษาบาลี ดังนี้

พยัญชนะวรรค ก

อักษรไทย	ก	ข	ค	ฆ	ง
อักษรธรรม	ᨠ	ᨡ	ᨣ	ᨤ	ᨥ

พยัญชนะวรรค จ

อักษรไทย	จ	ฉ	ช,ซ	ฌ	ญ
อักษรธรรม	ᨧ	ᨨ	ᨩ	ᨪ	ᨬ

พยัญชนะวรรค ฏ

อักษรไทย	ฏ	ฐ	ฑ	ฒ	ณ
อักษรธรรม	ᨭ	ᨮ	ᨯ	ᨰ	ᨱ

พยัญชนะวรรค ต

อักษรไทย	ต	ด	ถ	ท	ธ	น
อักษรธรรม	ᨾ	ᨿ	ᩁ	ᩅ	ᩃ	ᩈ

พยัญชนะวรรค ป

อักษรไทย	บ,ป	ผ	ฝ	พ	ฟ	ภ	ม
อักษรธรรม	ᨻ	ᨼ	ᨽ	ᨿ	ᩁ	ᩅ	ᩃ

พยัญชนะเศษวรรค

อักษรไทย	ย	ร	ล	ว	ส,ศ,ษ
อักษรธรรม	ᩃ	ᩅ	ᩈ	ᩅ	ᩈ

อักษรไทย	ห	ฬ	อ	ฮ
อักษรธรรม	ᩅ	ᩁ	ᩅ	ᩁ

2) *รูปพยัญชนะตัวเฟื้อง* หรือตัวสะกด คือ พยัญชนะที่เขียนเพียงครึ่งตัวหรือครึ่งรูป โดยส่วนใหญ่จะเขียนไว้ใต้บรรทัด ยกเว้นบางตัวเขียนข้างหน้าและข้างบนของพยัญชนะตัวเต็ม ทำหน้าที่เป็นตัวสะกดหรือควบกล้ำ ดังตัวอย่างรูปแบบและการใช้ต่อไปนี้

อักษร	รูปตัวเฟื้อง	การใช้
น	ᩈ	เรียกว่า ห้อย น หรือ ตีน น ใช้เขียนใต้บรรทัด เช่น ᩈᩅᩁᩈ (ขัน), ᩈᩅᩁᩈ (วัน)
ง	-᩠ᨿ	เรียกว่า ไม้อังแล่น ใช้เขียนบนตัวพยัญชนะ เช่น ᨠᩅᩁᩈᩅᩁᩈ (ของ), ᨠᩅᩁᩈᩅᩁᩈ (ห้อง), ᨠᩅᩁᩈᩅᩁᩈ (น้อง)
	-᩠ᨿ	เรียกว่า ห้อย ง หรือ ตีน ง ใช้เขียนใต้บรรทัด เช่น ᨠᩅᩁᩈᩅᩁᩈ (นาง), ᨠᩅᩁᩈᩅᩁᩈ (ทาง), ᨠᩅᩁᩈᩅᩁᩈ (ฟาง)
ม	-᩠ᨾ	เรียกว่าห้อย ม หรือตีน ม ใช้เขียนใต้บรรทัด เช่น ᨠᩅᩁᩈᩅᩁᩈᩅᩁᩈ

อักษร	รูปตัวเฟื้อง	การใช้
		(รัศมี)
บ,ป		เรียกว่า ห้อย ป หรือตีน ป ใช้เขียนหลังพยัญชนะ เช่น (กับ), (นับ), (สับ)
ย		เรียกว่า ห้อย ย หรือตีน ย ใช้เขียนหลังพยัญชนะ เช่น (ด้วย), (ควาย)
ร		เรียกว่า ห้อย ร หรือตีน ร ใช้เขียนหน้าพยัญชนะ เช่น (พระ), (สระ), (ประมาณ)
ล		เรียกว่า ห้อย ล หรือตีน ล ใช้เขียนใต้บรรทัด เช่น (หลาน)
ว		เรียกว่า ห้อย ว หรือ ตีน ว ใช้เขียนใต้บรรทัด เช่น (แล้ว), (แนว)
ส		เรียกว่า ห้อย ส หรือตีน ส ใช้เขียนใต้บรรทัด อย่างเช่นคำว่า ราช เขียนคำภาษาถิ่น แสดงเป็นตัวสะกดเฉพาะแม่กดเท่านั้น เช่น (มหาราช), (อากาศ)
		เรียกว่า ส สองห้อง (สฺส) ใช้เขียนบนบรรทัด เช่น (ตัสสะ), (รัสสะ)
อ		เรียกว่า ห้อย อ หรือตีน อ ใช้เขียนใต้บรรทัดแสดงหน้าที่เป็น สระ ออ เช่น (สอง), (น้อง), (จอง)
จ		เรียกว่า ห้อย จ หรือตีน จ ใช้เขียนใต้บรรทัด เช่น (สัจจะ)
ฐ		เรียกว่า ห้อย ฐ หรือตีน ฐ ใช้เขียนใต้บรรทัด เช่น (รัฏฐะ)
ถ		เรียกว่า ห้อย ถ หรือตีน ถ ใช้เขียนใต้บรรทัด เช่น (วัตถุ)
พ		เรียกว่า ห้อย พ หรือตีน พ ใช้เขียนใต้บรรทัด เช่น

อักษร	รูปตัวเฟื้อง	การใช้
		(บุพพะ)
ธ		เรียกว่า ห้อย ธ หรือตีน ธ ใช้เขียนใต้บรรทัด เช่น (พุทธะ)
ญ		ไม่มีรูปการเขียนใต้บรรทัด จะเขียนบนบรรทัดโดยมีความหมายเท่ากับ ญญ เช่น (สัญญา), (ปัญญา) ซึ่งหากเขียน หรือ จะผิด
ก		เรียกว่า ห้อย ก หรือตีน ก ใช้เขียนใต้บรรทัด มี ๒ แบบ เช่น (กกแนน) (นก)

2.2.2 รูปสระ ประกอบด้วย

1) *สระลอย* คือ สระที่สามารถเขียนและออกเสียงเองได้เลย ไม่ต้องอาศัยพยัญชนะอื่นใดมาประกอบ ซึ่งทำหน้าที่เป็นตัว อ และสระ มีอยู่ 8 ตัวดังนี้

อะ	อา	อิ	อี	อุ	อู	เอ	เอา

ตัวอย่างการใช้สระลอย

(อีสาน) (อุดม)

2) *สระจม* คือ สระที่ใช้เขียนผสมกับตัวพยัญชนะเท่านั้นถึงจะออกเสียงและใช้ได้ โดยสามารถเขียนได้รอบตัวพยัญชนะตัวเต็ม ดังตัวอย่างต่อไปนี้

สระไทย	-ะ	-า	อิ	อี	อึ	อื	อุ	อู
สระธรรม								
สระไทย	เ-ะ	เ-	แ-ะ	แ-	โ-ะ	โ-	อำ	
สระธรรม								

สระไทย	ไ/ใ	เ-า	เอีย	เอือ	เ-อ	อัว		
สระธรรม								
สระไทย	เ-าะ	ออ	ออย					
สระธรรม								

3) *สระพิเศษ* มี 5 ตัว ดังนี้

ตัวที่	รูปอักขระ	การใช้
1		นิคหิต ใช้เป็นตัว ง สะกดเมื่อเขียนเป็นคำภาษาบาลี และใช้แทนเสียง ออ เช่น (อิทัง) (ขอ)
2		ไม้กง หรือไม้ก่ม ใช้เป็นสระโอะลดรูป เช่น (คน) (หัว)
3	-□	ตัว หยอหยาดน้ำ ใช้เขียนเมื่อเป็นสระออ ที่มีตัวสะกดเป็น ย เช่น (น้อย) (คอย) (ซ่อย, ช่วย)

ตัวที่	รูปอักขระ	การใช้
4		สระอายาว ใช้เขียนกับตัว ว ( ) เพื่อไม่ให้ปนกับตัว ต ( ) เช่น ดู

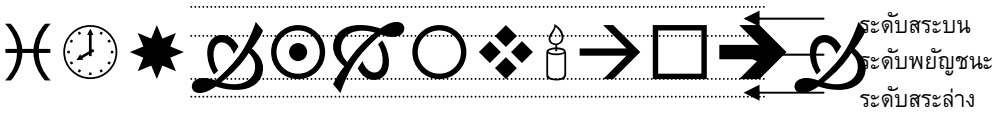
		คล้ายๆ กัน ท่านจึงใช้ (วา หรือ ว่า) เพื่อให้ดูง่ายขึ้น
5		สระ ฤา ใช้เขียนเหมือน ฤ ในภาษาไทย เช่น ฤาชา ก็จะเขียนเป็น

2.2.3 เลขอักษรธรรม ตัวเลขของอักษรธรรมอีสานมี 10 รูปเหมือนเลขไทย ดังนี้

เลขไทย	๑	๒	๓	๔	๕	๖	๗	๘	๙	๐
เลขธรรม	⇧	⇩	⇔	⇳	⇖	⇗	⇙	⇘	▫	⇨

เมื่อนำพยัญชนะและสระมาเขียนรวมกันเป็นคำหรือข้อความ จะมีลักษณะการเขียนเป็น 3 ระดับ (ดังภาพที่ 2.2) ดังนี้

1) ระดับสระบน (Upper vowel level)
2) ระดับพยัญชนะ (Consonant line level)
3) ระดับสระล่าง (Lower vowel line level)



ภาพที่ 2.2 แสดงระดับของอักษรธรรมอีสาน

2.3 การรู้จำรูปแบบ (Pattern Recognition)

การรู้จำรูปแบบสามารถแบ่งออกเป็น 2 ประเภท ดังนี้ [5]

2.3.1 การรู้จำในสิ่งที่เป็นรูปธรรม (The Recognition of Concrete Items)

คือ การรู้จำในสิ่งที่มีรูปร่าง สามารถสัมผัสหรือมองเห็นได้ด้วยตา การรู้จำประเภทนี้สามารถจำแนกย่อยเป็น 2 แบบคือ

1) การรู้จำในสิ่งที่มีรูปแบบอยู่อย่างถาวร เช่น วัตถุต่าง ๆ หรือตัวอักขระ เป็นต้น
2) การรู้จำในสิ่งที่มีรูปแบบอยู่ชั่วขณะหนึ่ง เช่น คลื่นเสียง อนุกรมเวลา เป็นต้น

2.3.2 การรู้จำในสิ่งที่เป็นนามธรรม (The Recognition of Abstract Items)

คือ การรู้จำในสิ่งที่ไม่มีรูปร่าง หรือเป็นการรู้จำในสิ่งที่ได้จากความคิด (Conceptual -Recognition) เช่น ความสามารถในการรู้จำเหตุผลเก่า ๆ หรือการหาคำตอบของปัญหาได้โดยไม่ต้องนำความสามารถทางตา หู ลิ้น หรือจมูกมาใช้งาน

2.4 ชนิดและคุณสมบัติของการรู้จำ

2.4.1 ชนิดของการรู้จำ จำแนกตามลักษณะการนำเข้าของข้อมูลเข้าสู่ระบบ ชนิดของการรู้จำสามารถจำแนกตามลักษณะของการนำเข้าของข้อมูลเข้าสู่ระบบได้ดังนี้

1) แบบ Off-line กระบวนการรู้จำจะกระทำหลังจากการเขียนอักขระหรือข้อความเสร็จสิ้น โดยผู้เขียนทำการเขียนตัวอักขระลงบนกระดาษจากนั้นจึงทำการสแกนเพื่อแปลงข้อมูลจากแผ่นกระดาษให้เป็นข้อมูลดิจิตอลที่มีลักษณะรูปภาพ 2 มิติ เรียกว่าไม่มีการประมวลในขณะเขียนและระบบจะมีการตอบสนองต่อผู้ใช้น้อยมาก จากนั้นจึงส่งข้อมูลดิจิตอลไปยัง input ของระบบรู้จำ

จุดเด่น สามารถนำไปประยุกต์ใช้กับข้อมูลนำเข้าที่มีขนาดใหญ่ได้เป็นอย่างดี จึงเหมาะกับงานที่มี ข้อมูลนำเข้าจำนวนมาก

2) แบบ On-line กระบวนการรู้จำจะกระทำในขณะที่กำลังเขียนอักขระหรือข้อความจนกระทั่งการเขียนเสร็จสิ้น บางครั้งเรียกว่า แบบ Real time หรือแบบ Dynamic โดยขณะที่มีการใช้งานระบบจะมีการตอบ สนองกับ Strokes[2] ของการเขียนตัวอักขระของผู้ใช้ในระหว่างที่มีการเขียน

จุดเด่น ใช้กับข้อมูลนำเข้าที่มีขนาดเล็กโดยมีการประมวลแบบทันทีทันใด (Real Time) เหมาะกับงานที่ต้องการความรวดเร็ว

2.4.2 ชนิดของการรู้จำ จำแนกตามวิธีการรู้จำ ชนิดของการรู้จำสามารถจำแนกตามวิธีการรู้จำได้ 4 ชนิดดังนี้

1) วิธีการเทียบคู่รูปแบบ

วิธีการนี้เป็นวิธีแรก ๆ ที่ใช้ในการรู้จำตัวอักขระ หลักการโดยทั่วไปคือสร้างต้นแบบขึ้นมาจากข้อมูลอักขระต่าง ๆ ปกติจะเป็นข้อมูล 2 มิติ โดยมีการกำหนดตำแหน่งที่สำคัญที่สามารถใช้แยกแยะความแตกต่างระหว่างตัวอักขระแต่ละตัว การทำงานของระบบจะนำรูปภาพที่ต้องการไปเปรียบเทียบกับต้นแบบเพื่อวัดความคล้ายคลึงกับข้อมูลต้นแบบ จากนั้นจึงระบุว่าเป็นรหัสของตัวอักขระใด วิธีการนี้ค่อนข้างอ่อนไหวต่อข้อมูลแทรกซ้อน ขนาดของภาพและการเอียงของตัวอักขระ

2) วิธีการทางสถิติ

วิธีนี้เป็นการใช้ค่าความน่าจะเป็น มาใช้ในการตัดสินใจรูปภาพนำเข้า ที่ได้มาจากขั้นตอนการสกัดลักษณะเฉพาะ และจะส่งเข้าไปในส่วนการรู้จำเฉพาะของตัวอักขระแต่ละตัว ซึ่งผลลัพธ์ที่ได้จากส่วนของการรู้จำทุกตัวจะเป็นค่าความน่าจะเป็นที่จะถูกนำมาใช้ในการเปรียบเทียบและตัดสินใจว่าข้อมูลที่เข้ามาเป็นตัวอักขระใด วิธีการนี้แต่ละต้นแบบจะถูกแทนด้วยจำนวนข้อมูลที่เป็นค่าตัวแทนจากการสกัดลักษณะสำคัญ ซึ่งจะเป็นชุดตัวเลขที่มีค่าสัมประสิทธิ์ต่าง ๆ กันตามที่ผู้ใช้กำหนด

3) วิธีการวิเคราะห์ทางโครงสร้าง

วิธีการนี้เป็นการนำภาพตัวอักขระแต่ละตัวมาวิเคราะห์โครงสร้าง โดยถือว่าตัวอักขระทุกตัวประกอบด้วยองค์ประกอบพื้นฐานที่ได้มาจากการสกัดลักษณะเฉพาะโดยการวิเคราะห์ทางโครงสร้าง

---

[2] **Strokes** หมายถึง ช่วงเวลาที่วางปลายปากกาลงจนกระทั่งยกปลายปากกาขึ้น

ซึ่งมักจะใช้ชื่อหรือค่าที่บอกว่าลักษณะโครงสร้างสำคัญนั้นเป็นอะไร เช่น เส้นตรง วงกลม เป็นต้น แทนที่จะเป็นค่าจำนวนจริง และในขั้นตอนการรู้จำลักษณะเฉพาะทั้งหลายที่ประกอบกันขึ้นเป็นตัวอักขระนั้นจะถูกส่งเข้าไปให้ส่วนตรวจวิเคราะห์กฎการเขียนตัวอักขระนั้น เช่น Formal Grammar Machine โครงสร้างกราฟ หรือโครงสร้างต้นไม้ เป็นต้น เพื่อระบุว่าเป็นตัวอักขระใด ซึ่งการตัดสินใจจะพิจารณาจากรูปแบบการเชื่อมต่อขององค์ประกอบต่าง ๆ ของตัวอักขระนั้น ๆ วิธีการนี้มีข้อดีคือความยืดหยุ่นต่อความหลากหลายของตัวอักขระค่อนข้างมาก แต่อย่างไรก็ตามอัตราความถูกต้องของวิธีนี้ขึ้นอยู่กับการสร้างกฎ และการวิเคราะห์กฎที่มีประสิทธิภาพ

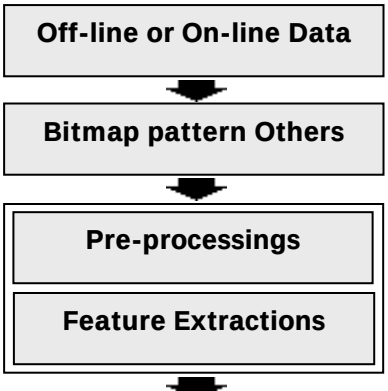
4) วิธีทางโครงข่ายประสาทเทียม

วิธีการนี้เป็นเทคนิคที่พยายามเลียนแบบการทำงานของสมองมนุษย์ ที่มีโครงข่ายเชื่อมต่อกันของหน่วยความจำย่อยจำนวนมากที่สะสมความรู้เอาไว้ ความรู้เหล่านี้จะได้จากการฝึกสอนไว้ก่อน เช่น การสอนให้รู้จักตัวอักขระ โดยส่งภาพของอักขระเข้าไปให้ระบบเรียนรู้พร้อมทั้งบอกว่ามีค่าเป็นรหัสของตัวอักขระใด โครงข่ายประสาทเทียมจะเรียนรู้ถึงรูปแบบตัวอักขระที่หลากหลายเพื่อความสามารถในการรู้จำภาพตัวอักขระหลาย ๆ รูปแบบ สิ่งที่สอนให้กับโครงข่ายประสาทเทียมไม่จำเป็นต้องเป็นรูปของตัวอักขระเสมอไปโดยข้อมูลอาจได้จากขั้นตอนการสกัดลักษณะเฉพาะและกระบวนการประมวลผลเบื้องต้นอื่น ๆ ก็ได้

2.5 กระบวนการรู้จำทางคอมพิวเตอร์

กระบวนการรู้จำอักขระมีขั้นตอนที่สำคัญคือ การประมวลผลเบื้องต้น (Pre-processing) การวิเคราะห์เพื่อสกัดลักษณะเฉพาะของตัวอักขระ (Feature extractions) และการจำแนกตัวอักขระ (Classifications) ดังภาพที่ 2.3 โดยในแต่ละขั้นตอนมีรายละเอียดดังนี้

2.5.1 Data เป็นข้อมูลตัวอักขระสำหรับนำเข้าไปประมวลผล ซึ่งมีหลายรูปแบบตามความต้องการของระบบ เช่น ในระบบที่ใช้ข้อมูลแบบออฟไลน์การเก็บข้อมูลจะทำได้โดยให้ผู้เขียนทำการเขียนตัวอักขระลงบนกระดาษจากนั้นจึงทำการสแกนเพื่อแปลงข้อมูลจากแผ่นกระดาษให้เป็นข้อมูลอิเล็กทรอนิกส์ที่มีลักษณะรูปภาพ 2 มิติ เรียกว่าไม่มีการประมวลในขณะเขียน (Airphaiboon et al., 1994) หรือในระบบที่ใช้ข้อมูลแบบออนไลน์ตัวข้อมูลจะได้จากการเก็บข้อมูลในตำแหน่งที่ปากกาเคลื่อนที่ไปบนแผ่นกระดานอิเล็กทรอนิกส์หรืออุปกรณ์ที่ใช้รับข้อมูล เช่น Tablet หรือ PDA (Tappert et al., 2004)



ภาพที่ 2.3 กระบวนการรู้จำ

2.5.2 Pre-processing เป็นการประมวลผลเบื้องต้นเพื่อปรับเปลี่ยนลักษณะรูปแบบบางประการของข้อมูลนำเข้า ทั้งนี้เพื่อปรับข้อมูลนำเข้าให้มีความเหมาะสมและตรงตามที่ระบบต้องการ เช่น การปรับขนาดของข้อมูลเป็น 32x32 หรือ 64x64 จุด การลดความหนาของเส้น (Thinning) ให้มีความหนา 1 จุด การกำจัดเส้นส่วนเกิน การปรับการกระจายความหนาแน่นของจุด (Dot density equalization)

2.5.3 Feature Extractions เป็นขั้นตอนการสกัดเอาลักษณะเฉพาะของตัวอักขระแต่ละตัวที่นำเข้าให้ออกมาเป็นข้อมูลเวกเตอร์เพื่อนำไปใช้เป็นตัวข้อมูลในการฝึกฝนและทดสอบระบบ ซึ่งการสกัดลักษณะเฉพาะของอักขระได้มีนักวิจัยเสนอวิธีการหลายวิธีการ เช่น การแบ่งอักขระออกเป็นบล็อกจำนวน 9 บล็อกแล้วคำนวณหาจำนวนจุดที่เป็นสีดำในแต่ละบล็อก (Pornchaikajornsak and Thammano, 2003) การใช้วิธีแปลงฟูริเยร์จากขอบของตัวอักขระเพื่อนำตัวบรรยายลักษณะฟูริเยร์ไปทำการเรียนรู้ในกระบวนการรู้จำ (โอฬาริก, 2546) การใช้วิธี Stroke Segmentation เพื่อแบ่งเส้นของอักขระออกเป็นส่วนย่อย (Sub-Stroke) จากนั้นจึงสกัดเอาลักษณะ เฉพาะออกมา เช่น จุดศูนย์กลางของ Sub-Stroke, มุมจากจุดเริ่มต้นถึงจุดสิ้นสุดของ Sub-Stroke หรือความยาวของ Sub-Stroke เป็นต้น (Budsayaplakorn et al., 2003) การพิจารณาถึงคุณลักษณะเด่น (Distinctive Feature) ของอักขระแต่ละตัวเช่น จำนวนของ Sub-Stroke จำนวนหัวอักขระ ระดับความสูงของหัวอักขระ สัดส่วนของความกว้างและความสูงของอักขระ (Choruengwiwat et al., 1998) เป็นต้น

2.5.4 Recognition เป็นขั้นตอนในการจำแนกและตัดสินใจว่าข้อมูลที่นำเข้ามานั้นเป็นตัวอักขระตัวใด ในขั้นตอนนี้มีนักวิจัยเสนอวิธีการหลายวิธีด้วยกัน เช่น วิธีการเปรียบเทียบชุด Feature Vector ของอักขระที่ต้องการทดสอบกับชุด Feature Vector ในฐานข้อมูลเพื่อค้นหาชุดของ Feature Vector ที่มีค่าใกล้เคียงมากที่สุดเรียกว่า Pattern Matching (Yamada et al., 1988) การใช้วิธี Rule-Base สร้างกฎต่าง ๆ ขึ้นมาเพื่อเปรียบเทียบ (Thongkamwitoon et al., 2002) หรือการใช้ตัวแบบฮิดเดนมาร์คอฟซึ่งเป็นตัวแบบทางสถิติแบบหนึ่งที่ถูกนำไปใช้ในการรู้จำเสียงและยังสามารถนำมาใช้ในการรู้จำอักขระได้อย่างมีประสิทธิภาพ (Rabiner, 1989)

2.6 ขั้นตอนวิธีลดความหนาของตัวอักขระ (Thinning Algorithm)

การลดความหนาของตัวอักขระ คือ การเปลี่ยนรูปร่างของภาพตัวอักขระ โดยการลดจำนวนเซตของจุดสีดำที่ติดกัน ทั้งนี้เพื่อให้ผลลัพธ์เป็นเส้นที่มีความหนา 1 จุด และผลลัพธ์ที่ได้จะเรียกว่า Skeleton โดยต้องรักษาโครงร่างพื้นฐานของภาพต้นฉบับที่เป็นข้อมูลนำเข้าไว้ให้ได้มากที่สุด ซึ่งการลดความหนาของเส้นจะช่วยลดความซับซ้อนของระบบ และลดขนาดของการใช้หน่วยความจำที่ใช้ในการเก็บโครงสร้างของรูปภาพที่จะนำมาประมวลผล ทำให้ในขั้นตอน Feature Extract ทำได้ง่ายและรวดเร็วยิ่งขึ้น

2.6.1 ประเภทของขั้นตอนวิธีการลดความหนา

ขั้นตอนวิธีการลดความหนาแบ่งออกเป็น 2 ประเภท ดังนี้

(1) **แบบอนุกรม** (Sequential or Serial Thinning Algorithm) ในการประมวลผลแบบอนุกรม ผลลัพธ์ของจุดใด ๆ ในการประมวลผลรอบที่ n จะขึ้นอยู่กับ neighbors ของจุดนั้นที่บางค่าอาจจะเป็นผลลัพธ์ที่เกิดจากการประมวลผลในรอบที่ n หรือ n-1 และวิธีการประมวลผลจะกระทำกับจุดทีละจุดจนครบ

(2) **แบบขนาน** (Parallel Thinning Algorithm) ในการประมวลผลแบบขนาน ผลลัพธ์ของจุดใด ๆ ในการประมวลผลรอบที่ n จะขึ้นอยู่กับ neighbors ของจุดนั้นที่เกิดจากรอบที่ n-1 เท่านั้น และการประมวลผลจะกระทำกับทุก ๆ จุดในแต่ละรอบ ซึ่งวิธีนี้เป็นวิธีที่นิยมใช้และเป็นวิธีที่ประมวลได้รวดเร็วกว่าการลดความหนาแบบอนุกรม

2.6.2 นิยามและข้อกำหนด

นิยาม รูปภาพ คือ เซตของจุดทุกจุดที่จะถูกพิจารณาและเป็นเซตที่มีขอบเขต (Finite set) โดยเซตจะประกอบด้วยจำนวนสมาชิกที่จำกัด

กำหนดให้ S เป็นรูปภาพที่มีขอบเขตจำกัดและ B = {0,1} โดยให้ Binary Image เป็นฟังก์ชั่น V ที่ใช้ในการกำหนด ค่า B ให้กับ S

กำหนดให้ ส่วนหน้า (Foreground) ของ Binary Image ประกอบด้วยจุดสีดำ คือ เซต P และส่วนของพื้นหลังประกอบด้วยจุดสีขาว คือ เซต P' ดังนั้นสามารถนิยาม P ได้ดังสมการที่ 1

$$P = \{p \mid p \in S \wedge V(p) = 1\} \qquad (1)$$

เมื่อ p คือ จุดใด ๆ

ทุกสมาชิกรอบจุด p จะเรียกว่า neighbors pixel และทั้งหมดจะอยู่ในขอบเขตของ 3x3 neighbor-hood ของจุด p คือ N(p) ดังภาพที่ 2.4

p[3]	p[2]	p[1]
p[4]	p	p[0]

p[5]	p[6]	p[7]

ภาพที่ 2.4 Neighborhood ของ N(p)

โดยภายใน 3x3 neighborhood จุด p คือ Break point (Connected point) ถ้าลบจุด p ออกไปจะทำให้ neighbors ของจุด p มีลักษณะเป็น disconnected

2.7 ตัวแบบฮิดเดนมาร์คอฟ (Hidden Markov -Models : HMMs)

ตัวแบบฮิดเดนมาร์คอฟเป็นเทคนิคการเรียนรู้ทางสถิติ (Statistical machine learning) ที่มีประสิทธิภาพสูง โดยได้พัฒนามาจากตัวแบบมาร์คอฟ (Markov Models) ซึ่งคิด ค้นขึ้นโดยนักคณิตศาสตร์ชาวรัสเซียชื่อ Andrei Andreevich Markov โดยใช้จำแนกรูปแบบของลำดับเหตุการณ์หนึ่งๆ ซึ่งลำดับของเหตุการณ์จะถูกจำลองให้อยู่ในรูปของลำดับการเปลี่ยนแปลงของสถานะ วิธีการนี้จะอาศัยค่าความน่าจะเป็นของการเปลี่ยนสถานะ และค่าความน่าจะเป็นของชนิดข้อมูล(Observation Symbol) ของตัวแบบเมื่อมีการเปลี่ยนสถานะเกิดขึ้น และผลลัพธ์จะถูกนำไปใช้ในการตัดสินใจว่าข้อมูลที่นำเข้ามาคืออะไร

2.7.1 องค์ประกอบของตัวแบบฮิดเดนมาร์คอฟ

องค์ประกอบที่สำคัญของตัวแบบฮิดเดนมาร์คอฟมีดังนี้ (Rabiner, 1989)

2.7.1.1 N คือ จำนวนของสถานะในตัวแบบ โดยที่เซตของสถานะเป็น S = { $S_1, S_2, \ldots, S_N$ }

(1) ให้ q_t เป็นสถานะที่เวลา t

(2) (Hidden) State sequence Q = { $q_1, q_2, \ldots, q_N$ }

โดยที่ $q_n \in$ S

2.7.1.2 M คือ จำนวนชนิดของข้อมูลในสถานะและแทนชนิดของข้อมูล เป็น V = { $V_1, V_2, \ldots, V_M$ } ซึ่งจะต้องสอดคล้องกับอินพุตที่ป้อนให้กับตัวแบบ

2.7.1.3 ความน่าจะเป็นของการเปลี่ยนสถานะ $A = \{a_{ij}\}$ โดยที่ $a_{ij} = P[q_{t+1} = S_j | q_t = S_i]$, เมื่อ $1 \le i, j \le N$

2.7.1.4 การกระจายความน่าจะเป็นของข้อมูลที่สถานะ j คือ $B = \{b_j(k)\}$

โดยที่ $b_j(k) = P[V_k \, at \, t | q_t = S_j]$

เมื่อ $1 \le j \le N$, $1 \le k \le M$

2.7.1.5 ความน่าจะเป็นของสถานะเริ่มต้น $\pi = \{\pi_i\}$ โดยที่ $\pi_i = P[q_1 = S_i]$, $1 \le i \le N$

จากค่า N, M, A, B และ π ในตัวแบบ ทำให้มีลำดับการสังเกต (Observation Sequence : O) เป็น $O = O_1, \ldots, O_T$ โดยที่ $O_t \in$ V และ T เป็นจำนวนของการสังเกตและสามารถบอกได้ว่าค่าต่าง ๆ เหล่านี้เป็นลักษณะเฉพาะของตัวแบบ ซึ่งตัวแบบสามารถแสดงได้เป็น λ = (A, B, π)

2.7.2 การสอนตัวแบบเพื่อการรู้จำตัวอักขระ

เนื่องจากตัวอักขระแต่ละตัวมีลักษณะเฉพาะไม่เหมือนกัน ดังนั้นจึงต้องสร้างตัวแบบฮิดเดนมาร์คอฟ ให้กับตัวอักขระแต่ละตัว ซึ่งต้องกำหนดค่าพารามิเตอร์เริ่มต้นให้กับตัวแบบแต่ละตัวแบบ แล้วคำนวณหาความน่าจะเป็นของลำดับการสังเกต (O) ต่อตัวแบบ หลังจากนั้นจึงทำการประมาณค่าตัวแปรเพื่อหาค่าพารามิเตอร์ใหม่ให้กับตัวแบบ ซึ่งจะเป็นการปรับค่าต่าง ๆ ของตัวแบบนั่นเอง ขั้นตอนทั้งหมดแสดงดังภาพที่ 2.5 และมีรายละเอียดของการคำนวณดังต่อไปนี้

2.7.2.1 การคำนวณหาค่าความน่าจะเป็นของอินพุตหรือลำดับการสังเกตต่อตัวแบบ ใช้หลักการ Forward-Backward Algorithm (Rabiner, 1989) ซึ่งมีรายละเอียดดังนี้

Forward Algorithm: กำหนดให้

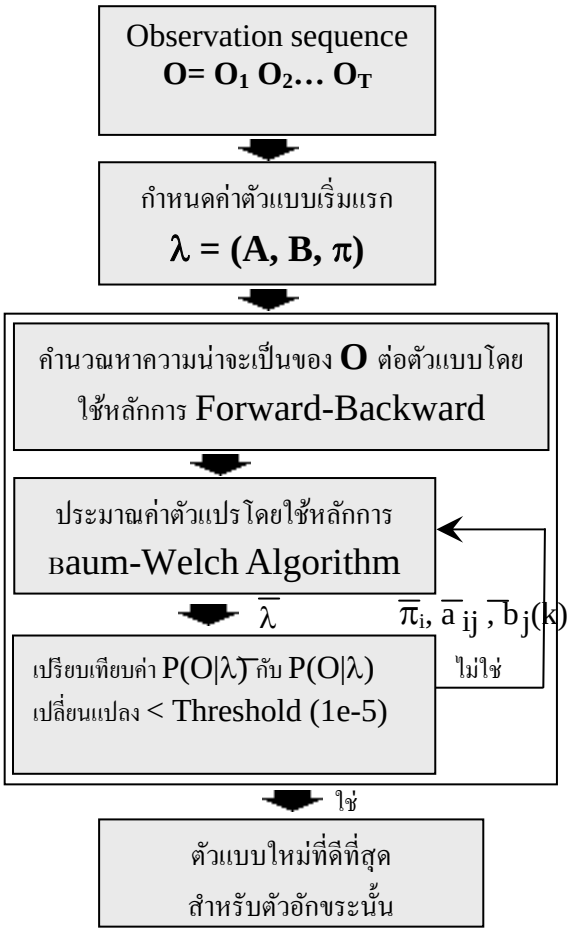
$\alpha_t(i)$ เป็นตัวแปร Forward ณ เวลา t ที่สถานะ i และกำหนดให้ $\alpha_t(i) = P(O_1, O_2 ... O_t, q_t = S_i | \lambda)$

π_i เป็นค่าความน่าจะเป็น ณ เวลา $t = 1$ ที่สถานะ i

$b_j(O_t)$ เป็นค่าความน่าจะเป็นของข้อมูล O ณ เวลา t ที่สถานะ j

(1). Initialization

$$\alpha_1(i) = \pi_i b_i(o_1) \ , \ 1 \le i \le N \qquad (1)$$



ภาพที่ 2.5 แสดงขั้นตอนการสอนตัวแบบ

(2). Induction

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij} \right] b_j(o_{t+1}) \ , \qquad (2)$$

$$1 \le t \le T-1 \ , \ 1 \le j \le N$$

(3). Termination

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i) \qquad (3)$$

Backward Algorithm: กำหนดให้

$\beta_t(i)$ เป็นตัวแปร Backward ณ เวลา t ที่สถานะ i และกำหนดให้

$\beta_t(i) = P(O_{t+1}, O_{t+2}, ..., O_T | q_t = S_i, \lambda)$

π_i เป็นค่าความน่าจะเป็น ณ เวลา $t = 1$ ที่สถานะ i

$b_j(O_t)$ เป็นค่าความน่าจะเป็นของข้อมูล O ณ เวลา t ที่สถานะ j

(1). Initialization

$$\beta_T(i) = 1 \ , \ 1 \le i \le N \qquad (4)$$

(2). Induction

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j) \ , \qquad (5)$$

$$T-1 \le t \le 1 \ , 1 \le i \le N$$

(3). Termination

$$P(O|\lambda) = \sum_{i=1}^{N} \pi_i b_i(O_1) \beta_1(i) \qquad (6)$$

2.7.2.2 การประมาณค่าตัวแปร (Parameter estimation) ใช้หลักการของ Baum-Welch (Rabiner, 1989) โดยนำผลลัพธ์ที่ได้จาก Forward-Backward Algorithm มาใช้ ซึ่งมีรายละเอียดดังนี้

(1). กำหนดให้ $\xi_t(i, j)$ คือความน่าจะเป็นของการอยู่ในสถานะ *i* ที่เวลา *t* และสถานะ *j* ที่เวลา *t*+1

$$\xi_1(i, j) = \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)} \qquad (7)$$

(2). กำหนดให้ $\gamma_t(i)$ เป็นความน่าจะเป็นของการอยู่ในสถานะ *i* ที่เวลา *t* ซึ่งเกิดจากลำดับการสังเกตและตัวแบบ

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i, j) \qquad (8)$$

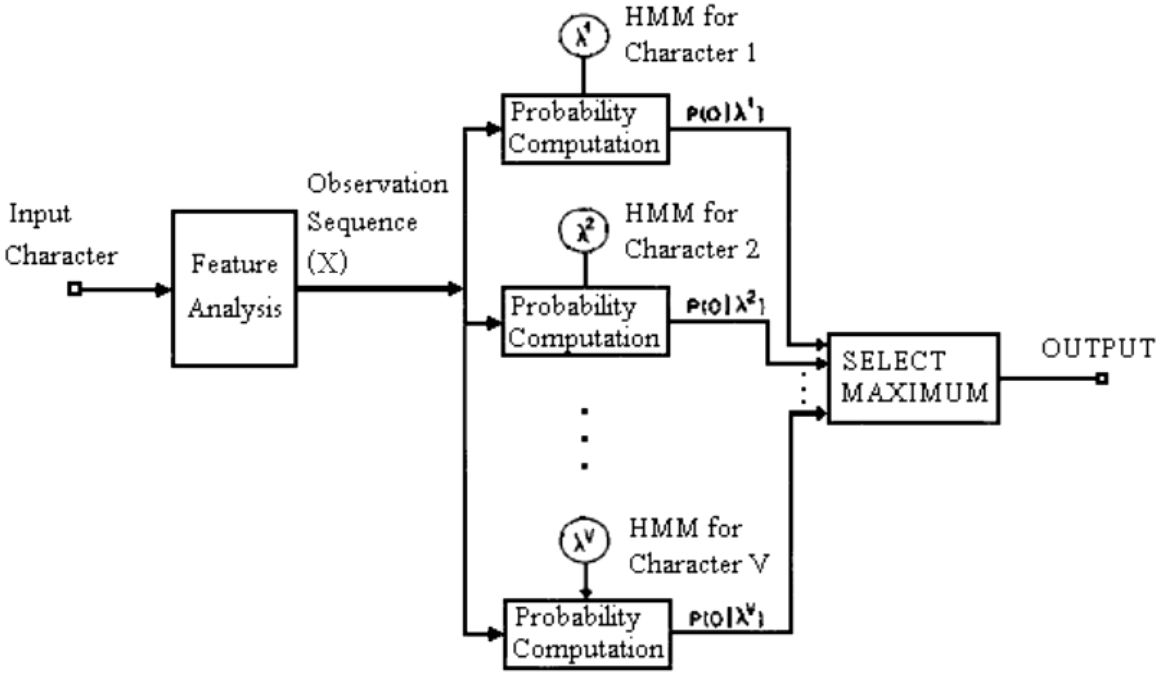
(3). การประมาณตัวแปรใหม่ (Reestimation) ของ A, B และ π คือ

$$\overline{\pi}_i = \gamma_1(i) \qquad (9)$$

$$\overline{a_{ij}} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \qquad (10)$$

$$\overline{b_j}(k) = \frac{\sum_{o_t = v_k}^{T} \gamma_t(j)}{\sum_{t=1}^{T} \gamma_t(j)} \qquad (11)$$

2.7.3 การนำตัวแบบฮิดเดนมาร์คอฟไปใช้ในการรู้จำตัวอักษรธรรมอีสาน ในการรู้จำโดยใช้ตัวแบบฮิดเดนมาร์คอฟ ทำได้โดยการคำนวณหาความน่าจะเป็นของลำดับการสังเกตต่อตัวแบบด้วยหลักการ Forward-Backward Algorithm โดยเมื่อตัวแบบได้รับค่าลำดับการสังเกตที่ถูกต้องแล้ว จะคำนวณได้ค่าความน่าจะเป็นสูงสุด โดยข้อมูลที่ป้อนเข้าไปในระบบจะถูกวิเคราะห์เพื่อสกัดเอาลักษณะเฉพาะแล้วนำมาคำนวณหาค่าความน่าจะเป็นสำหรับแต่ละตัวแบบ ซึ่งค่าความน่าจะเป็นสูงสุดที่คำนวณได้จากตัวแบบนั้นๆ จะเป็นค่าที่นำมาใช้ในการตัดสินใจว่าข้อมูลที่นำเข้ามานั้นเป็นตัวอักขระตัวใด ดังแสดงในภาพที่ 2.6

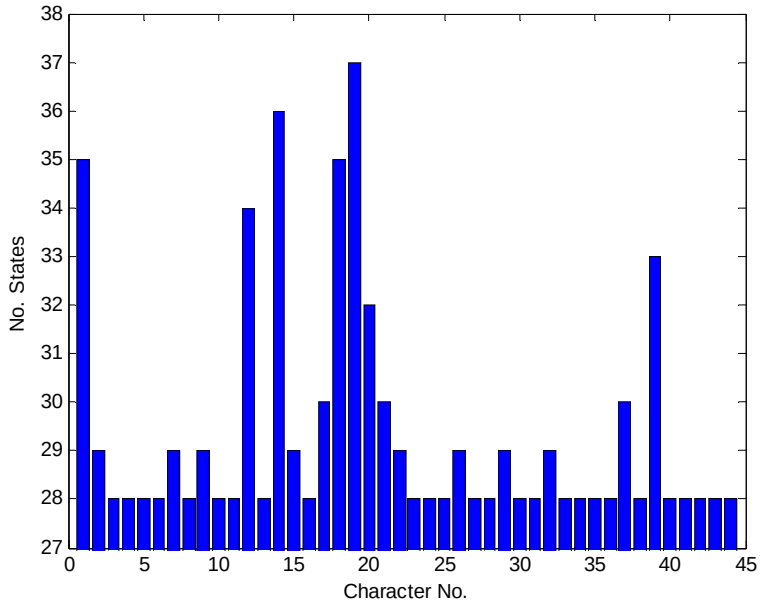


ภาพที่ 2.6 Block Diagram การรู้จำอักขระโดยใช้ตัวแบบฮิดเดนมาร์คอฟ

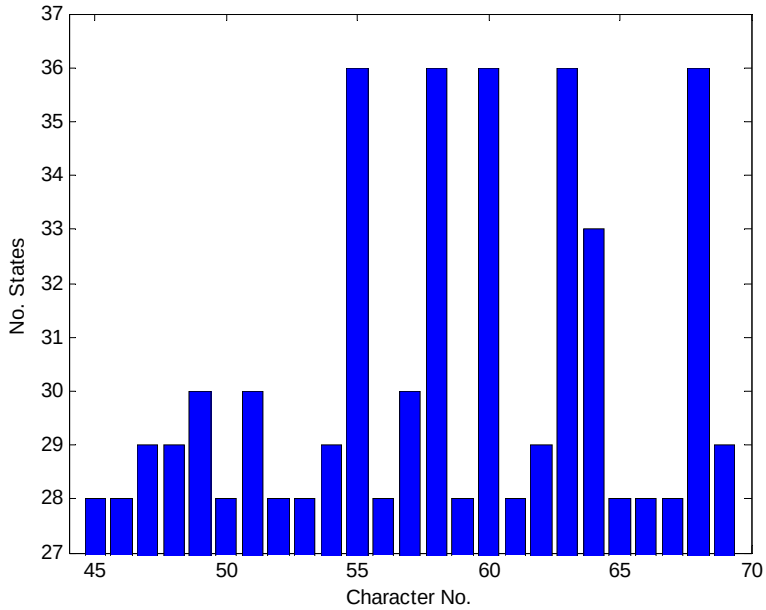
2.8 งานวิจัยที่เกี่ยวข้อง

2.8.1 สุรศักดิ์ ตั้งสกุล (2548) ได้ทำการศึกษาเรื่อง การรู้จำอักขระตัวเขียนภาษาไทยโดยใช้ตัวแบบฮิดเดนมาร์คอฟ โดยการสกัดลักษณะเฉพาะที่สำคัญด้วยวิธีการปรับการกระจายของจุดของตัวอักขระแบบไม่เชิงเส้นและคุณลักษณะเด่นของอักขระหลายวิธีร่วมกันโดยใช้ตัวแบบฮิดเดนมาร์คอฟชนิดต่างๆ เพื่อศึกษาเปรียบเทียบประสิทธิภาพ โดยมีการกำหนดจำนวนสถานะของตัวแบบในลักษณะที่แน่นอนและไม่แน่นอน ผลการวิจัยพบว่าจำนวนสถานะที่เหมาะสมและน้อยที่สุดที่ให้ผลอัตราการรู้จำเฉลี่ยสูงสุดสำหรับตัวแบบชนิด Left-Right มีจำนวนสถานะแน่นอน 35 สถานะ โดยมีอัตราการรู้จำอักขระในกลุ่มพยัญชนะเท่ากับ 90.07% และอักขระในกลุ่มสระ วรรณยุกต์ และอักขระพิเศษเท่ากับ 94.09% ในขณะที่ผลการทดสอบกับตัวแบบ 5 ชนิด ที่มีจำนวนสถานะไม่แน่นอน ให้ผลลัพธ์ในการรู้จำตัวอักขระกลุ่มพยัญชนะเฉลี่ยที่ดีที่สุดคือ 91.67% โดยใช้ตัวแบบ Left-Right Allow Skip State แบบที่ 2 (แสดงใน

ภาพที่ 2.7 ข.) ขณะที่ในกลุ่มตัวอักขระ สระ วรรณยุกต์ และอักขระพิเศษ ให้ผลลัพธ์โดยเฉลี่ยดีที่สุด คือ 96 % โดยใช้ตัวแบบชนิด Left-Right-Left Allow Skip State

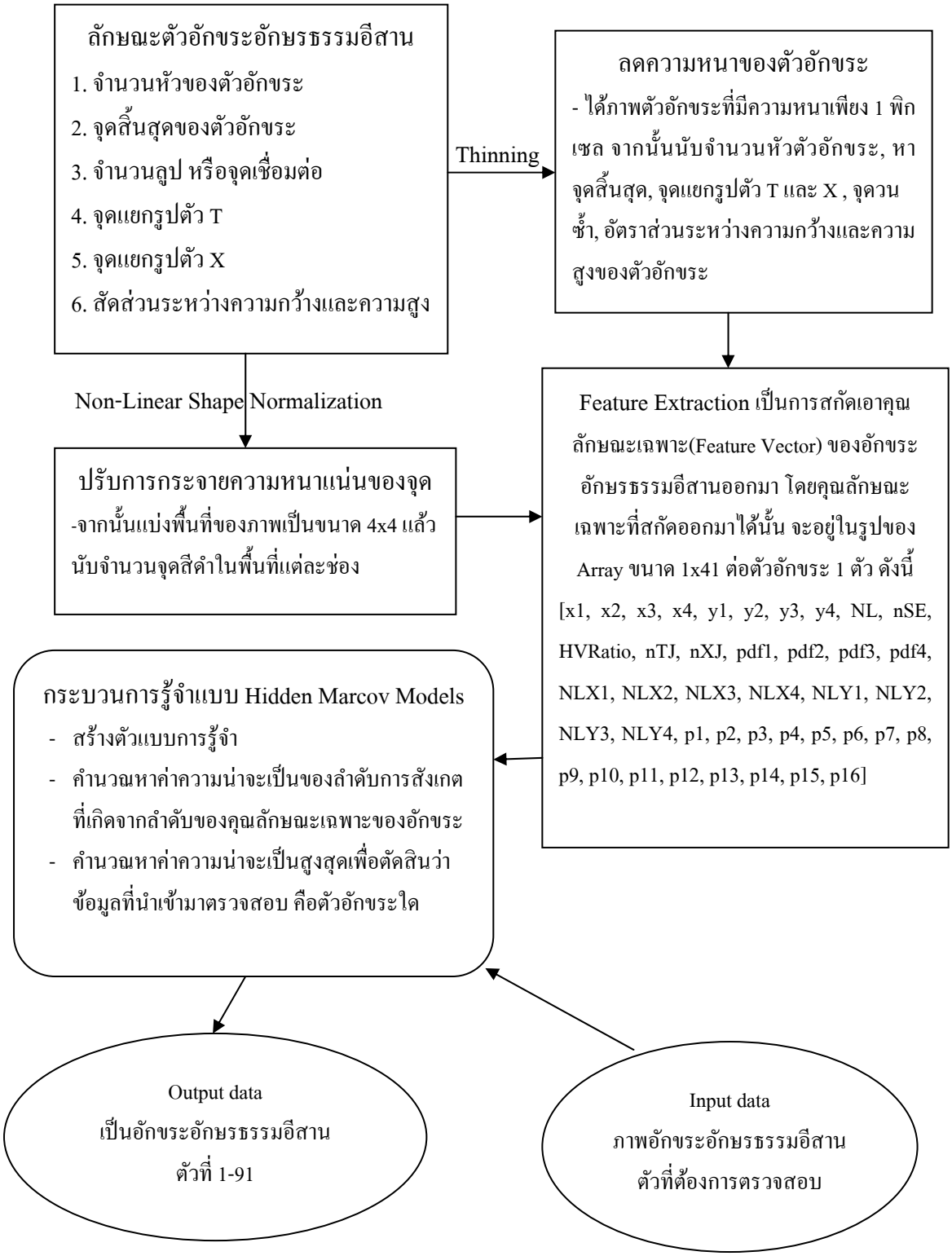


ภาพที่ 2.7 ก จำนวนสถานะที่เหมาะสมของตัวแบบชนิด Left-Right ของตัวอักขระกลุ่มที่ 1



ภาพที่ 2.7 ข จำนวนสถานะที่เหมาะสมของตัวแบบชนิด Left-Right ของตัวอักขระกลุ่มที่ 2

ส่วนการสอนตัวแบบในลักษณะที่ไม่กำหนดจำนวนสถานะที่แน่นอน โดยจะเลือกตัวแบบที่เหมาะสมจากการคำนวณค่าความน่าจะเป็นที่ได้จากตัวแบบที่มีค่าสูงที่สุดเป็นตัวแบบของอักขระนั้น ผลลัพธ์ของจำนวนสถานะของตัวแบบชนิด Left-Right ที่ถูกเลือกให้เป็นตัวแบบของตัวอักขระกลุ่มที่ 1 แสดงดังภาพที่ 2.7 ก.

2.9 กรอบแนวคิดการทำวิจัย

ในการทำวิจัย ได้วางกรอบแนวคิดการทำวิจัยไว้ดังนี้