บทที่ 3
กระบวนการรู้จำเสียงพูดตัวเลขไทย

วิทยานิพนธ์นี้ได้พัฒนาโปรแกรมสำหรับกระบวนการรู้จำเสียงพูดตัวเลขไทยในส่วนต่าง ๆ ดังนี้

- ส่วนของโปรแกรมตัดคำเพื่อหาจำนวนพยางค์ และตัดหัวท้ายคำ
- ส่วนของการนอร์แมลไลซ์ (Normalization)
- ส่วนของการคำนวณค่าสัมประสิทธิ์ LPC
- ส่วนของนิวรอลเน็ตเวิร์ก
- ส่วนของกฎเกณฑ์การตัดสินใจ

วิธีการดำเนินการวิจัย

บันทึกเสียงพูดคำว่า 0-9 และ 11,12,13,17,20,30,21,23,27,32,78,93 ของกลุ่มบุคคลที่ใช้เป็นเสียงต้นแบบและใช้ในการทดสอบ คนละ 3 ครั้ง จำนวน 30 คน เพื่อใช้ในการฝึก (training) นิวรอลเน็ตเวิร์ก(เสียงพูด 2 ครั้งแรกใช้ในการฝึก (training) เสียงพูดครั้งที่ 3 ใช้ในการทดสอบเบื้องต้น) และนำบุคคลกลุ่มใหม่อีกกลุ่มหนึ่ง มาบันทึกเสียงพูด คนละ 3 ครั้ง จำนวน 12 คน เพื่อใช้เป็นข้อมูลในการทดสอบเพื่อวัดประสิทธิภาพของโปรแกรม บุคคลทั้งหมดที่นำมาบันทึกเสียงพูดเป็นชายทั้งสิ้น เสียงพูดถูกบันทึกโดยการ์ด sound blaster pro โดยบันทึกอยู่ในรูปข้อมูลขนาด 8 บิต และใช้ค่าสุ่มสัญญาณ 8 KHz ในการสุ่มสัญญาณเสียง ซึ่งมีค่าสูงเพียงพอเพราะความถี่ของสัญญาณเสียงที่ฟังรู้เรื่องอยู่ในช่วง 200 ถึง 4000 Hz สัญญาณเสียงพูดทั้งหมดถูกปรับแต่งให้เหมาะสมโดยการตัดหัวท้ายคำ ซึ่งจะตัดส่วนที่ไม่ใช่เสียงพูดออกเพื่อให้ได้เสียงพูดที่ไม่มีสัญญาณรบกวน จากนั้นนำสัญญาณเสียงพูดไปผ่านขั้นตอนการนอร์แมลไลซ์ (normalization) เพื่อให้เสียงทุกคำมีช่วงเวลาเท่ากัน และขั้นตอนคำนวณค่าสัมประสิทธิ์ LPC ที่เป็นตัวแทนของเสียงพูดนั้น ค่าสัมประสิทธิ์ LPC ที่คำนวณได้จะถูกป้อนเป็นข้อมูลอินพุตให้กับนิวรอลเน็ตเวิร์กเพื่อใช้ในการฝึก (training) หรือใช้ในการทดสอบ ขั้นตอนต่าง ๆ ในการรู้จำเสียงมีรายละเอียดดังนี้

3.1 โปรแกรมตัดหัวท้ายคำและหาจำนวนพยางค์

เนื่องจากสัญญาณเสียงพูดที่ได้จากการบันทึกมีส่วนที่ไม่ใช่เสียงพูดอยู่ด้วย จึงต้องมีการตัดหัวท้ายคำเพื่อให้ได้ข้อมูลเฉพาะส่วนที่ต้องการ การตัดหัวท้ายคำทำได้โดยการแบ่งสัญญาณเสียงพูดทั้งหมดออกเป็นส่วนย่อยแต่ละส่วนมีข้อมูล 100 จุดเพื่อให้ง่ายต่อการคำนวณและการตรวจสอบแล้วหาค่าพลังงานของแต่ละส่วนย่อย

$$E_s = \sum_{n=1}^{100} s^2(n) \tag{3.1}$$

เพื่อลดเวลาที่ใช้ในการคำนวณในวิทยานิพนธ์นี้จึงเปลี่ยนมาใช้ค่าผลรวมของค่าสัมบูรณ์ของแอมพลิจูดในแต่ละส่วนย่อยแทนดังสมการ

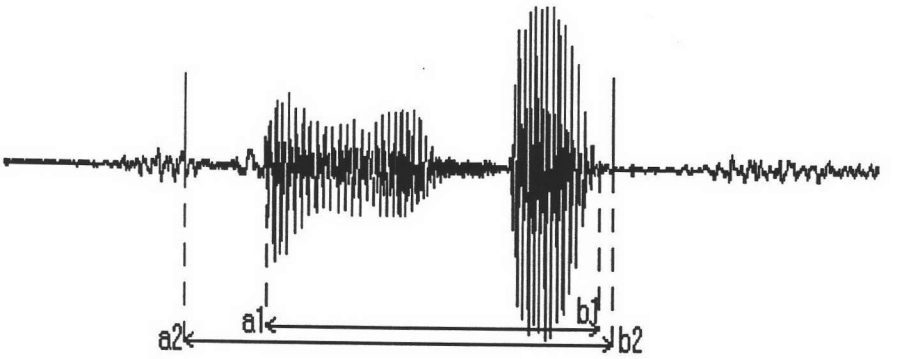
$$E_s = \sum_{n=1}^{100} |s(n)| \tag{3.2}$$

กำหนดให้ $E_{\max}$ เป็นค่าพลังงานในส่วนย่อยที่มีค่าพลังงานสูงสุดในเสียงพูดที่กำลังพิจารณา จากนั้นกำหนดระดับพลังงานอ้างอิง 2 ระดับเพื่อใช้ในการนับจำนวนพยางค์โดยข้อมูลที่นับว่าเป็นพยางค์จะต้องมีคุณสมบัติทั้ง 2 ข้อ ดังนี้

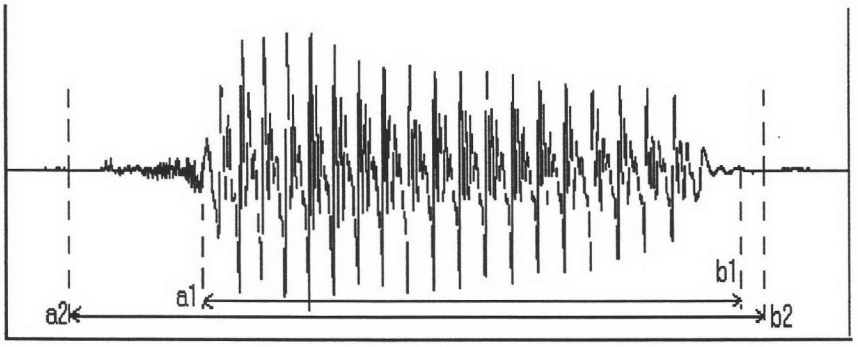
(1) มีส่วนย่อยที่มีพลังงานมากกว่า a เท่าของ $E_{\max}$ เป็นจำนวน m ส่วนย่อยติดกันขึ้นไป

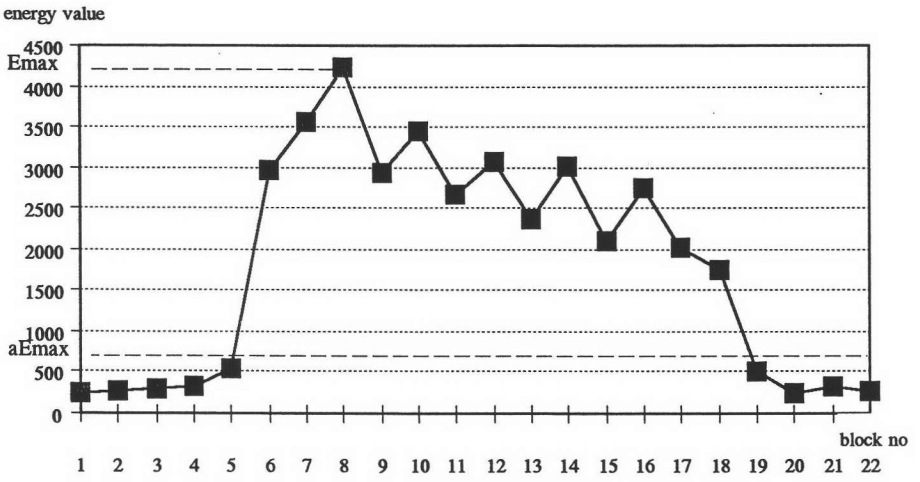
(2) มีส่วนย่อยที่มีพลังงานมากกว่า b เท่าของ $E_{\max}$ เป็นจำนวน n ส่วนย่อยติดกันขึ้นไป เมื่อ $b > a$ และ $m \geq n$

ค่าระดับพลังงานในคุณสมบัติข้อ (1) เป็นระดับพลังงานเริ่มต้นที่นับว่าเป็นพยางค์ ใช้เพื่อแยกส่วนที่เป็นเสียงพูดออกจากเสียงสภาพแวดล้อมในขณะบันทึกเสียง ส่วนค่าระดับพลังงานในคุณสมบัติข้อ (2) เป็นระดับพลังงานที่ยอมรับว่าเป็นพยางค์จริง ๆ ใช้เพื่อแยกเสียงพูดจากเสียงรบกวนที่มีระดับความดังสูง เช่น เสียงหายใจหรือเสียงเคาะไมโครโฟน ตัวอย่างของเสียงรบกวนประเภทนี้แสดงในรูปที่ 3.1 การใช้ระดับพลังงานเป็นจำนวนเท่าของพลังงานในส่วนย่อยที่มีค่ามากสุดทำให้โปรแกรมนี้สามารถทำงานได้ถูกต้องแม้ว่าเสียงพูดมีระดับความดังต่างกัน



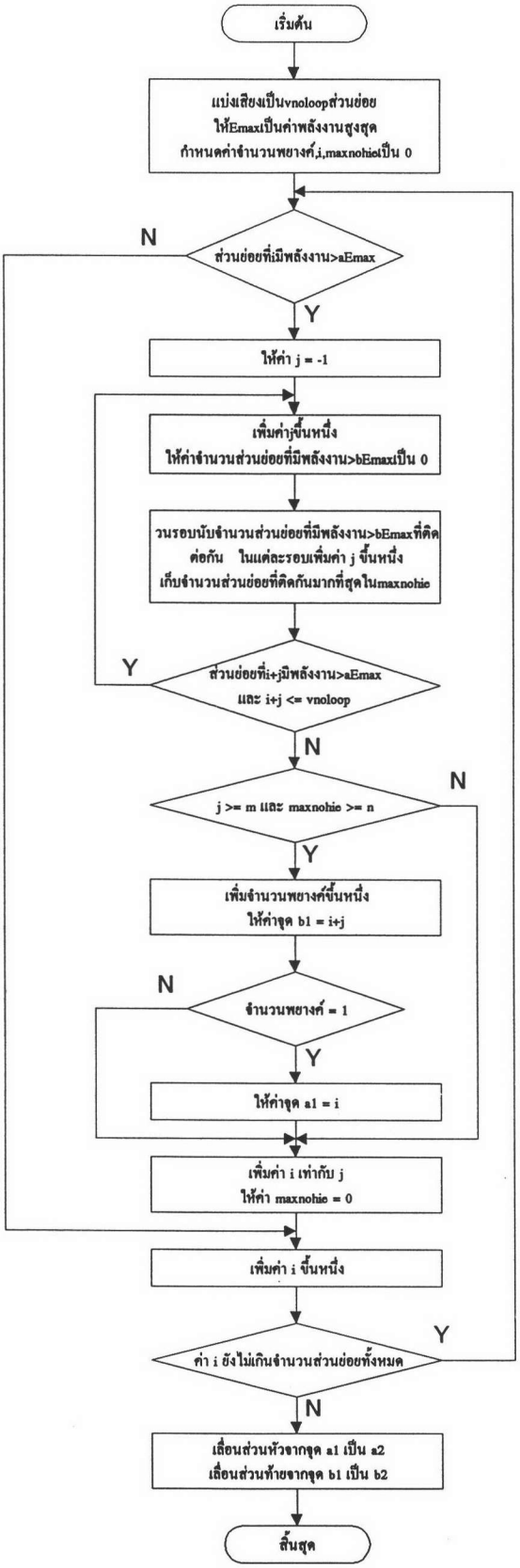
รูปที่ 3.1 รูปคลื่นของคำสองพยางค์ที่มีเสียงรบกวน





รูปที่ 3.2 รูปคลื่นและพลังงานของคำพยางค์เดียว

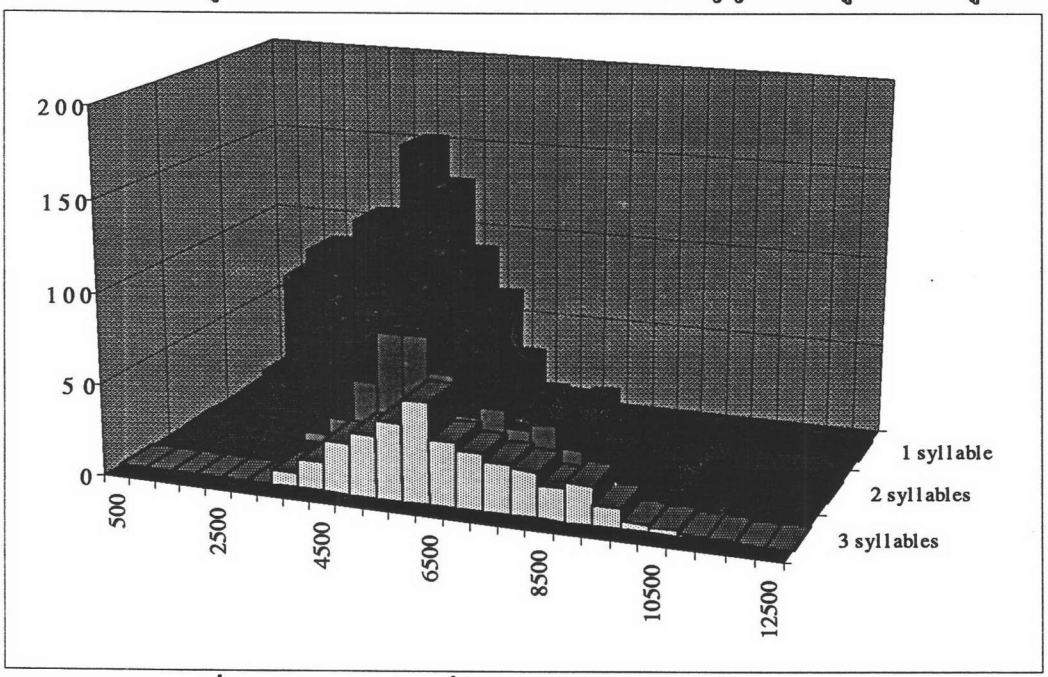
จากรูปที่ 3.2 ซึ่งเป็นรูปคลื่นของคำพยางค์เดียว ส่วนของคำจะเริ่มจากจุดเริ่มต้นของพยางค์แรกจนถึงจุดสิ้นสุดของพยางค์สุดท้ายที่ตรวจพบ นั่นคือระยะจากจุด a_1 ถึง b_1 จากนั้นทำการเลื่อนส่วนหัวและส่วนท้ายคำออกไปเท่ากับคาบเวลาคงที่ค่าหนึ่ง เพื่อให้ได้รายละเอียดของเสียงพูดส่วนต้นและท้ายคำ โดยที่คาบเวลาในการเลื่อนส่วนหัวอาจไม่เท่ากับคาบเวลาในการเลื่อนส่วนท้าย เสียงพูดจากส่วนหัวจนถึงส่วนท้ายคำคือระยะจากจุด a_2 ถึง b_2 ถูกใช้เป็นสัญญาณเสียงพูดที่จะทำการวิเคราะห์ต่อไป ส่วนสัญญาณเสียงพูดที่อยู่นอกช่วงนี้จะถือว่าเป็นเสียงของสภาพแวดล้อมและสัญญาณรบกวนซึ่งถูกตัดทิ้ง ขั้นตอนของการตัดหัวท้ายเสียงพูดตามที่อธิบายมานั้นแสดงในรูปที่ 3.3



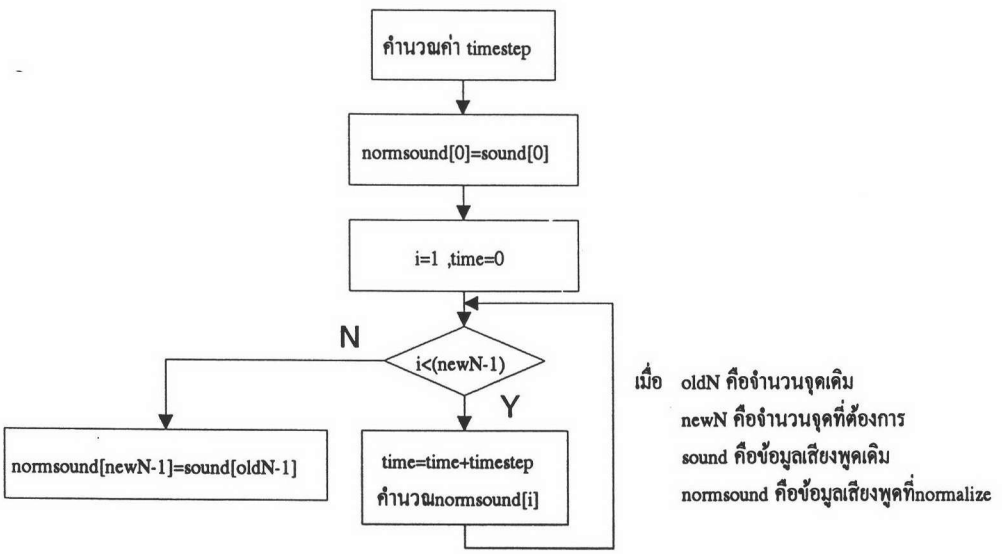
รูปที่ 3.3 ขั้นตอนของการตัดหัวท้ายเสียงพูด

3.2 การนอร์แมลไลซ์ (Normalization)

เนื่องจากสัญญาณเสียงพูดแต่ละคำมีความยาวไม่เท่ากัน แต่นิวรอลเน็ตเวิร์กมีจำนวนโหนดในระดับข้อมูลเข้า (input layer) คงที่ จึงต้องมีการนอร์แมลไลซ์สัญญาณเสียงพูดแต่ละคำ ให้มีช่วงเวลาเท่ากัน โดยใช้การประมาณค่าในช่วง (interpolation) ให้สัญญาณเสียงพูดยาวเท่ากับ 4,000 จุดข้อมูลสำหรับคำพยางค์เดียว,เท่ากับ 5,000 จุดข้อมูลสำหรับคำสองพยางค์ และเท่ากับ 6,000 จุดข้อมูลสำหรับคำสามพยางค์ ซึ่งเป็นช่วงเวลาที่มีข้อมูลจำนวนมากสุดสำหรับเสียงแบ่งตามจำนวนพยางค์ดังแสดงในรูปที่ 3.4 ขั้นตอนของการนอร์แมลไลซ์สัญญาณเสียงพูดแสดงในรูปที่ 3.5



รูปที่ 3.4 จำนวนเสียงพูดที่ความยาวต่าง ๆ ของเสียง 1,2 และ 3 พยางค์



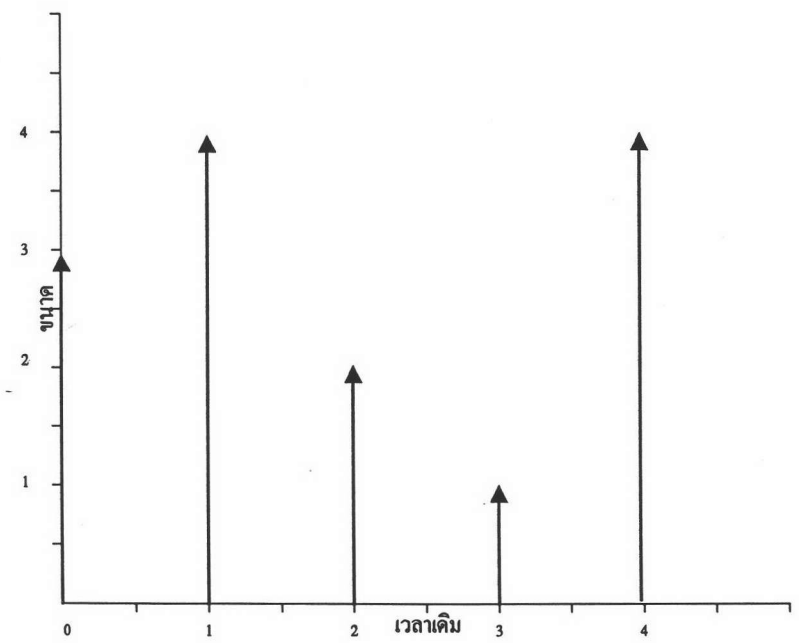
รูปที่ 3.5 ขั้นตอนของการนอร์แมลไลซ์สัญญาณเสียงพูด

ค่าต่าง ๆ ในรูปที่ 3.5 คำนวณได้ดังนี้

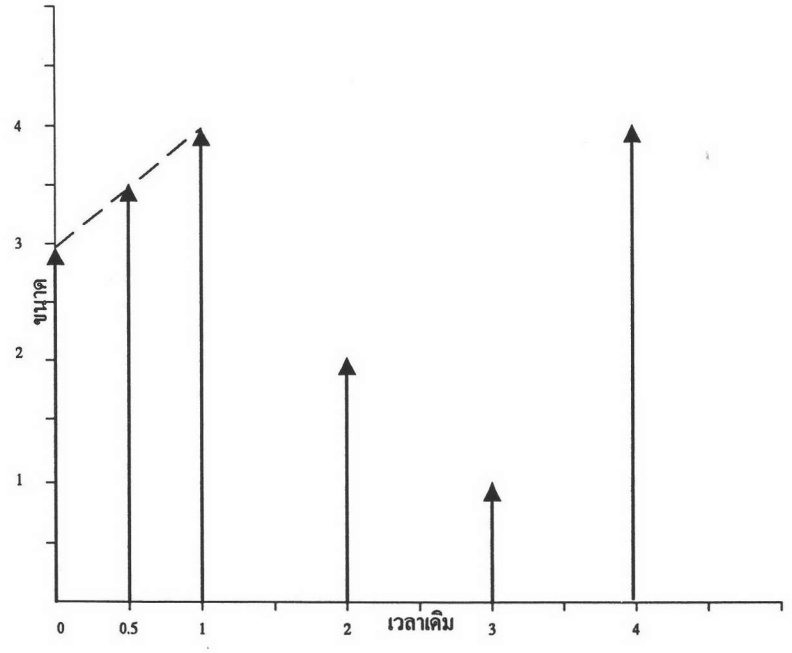
timestep = (จำนวนจุดข้อมูลเดิม-1)/(จำนวนจุดข้อมูลที่ต้องการ-1)

normsound[i] = sound[lowtime]+residue*(sound[lowtime+1]-sound[lowtime]) เมื่อ lowtime,residue คือจำนวนเต็มและเศษเหลือของค่า time ในแต่ละรอบของ i

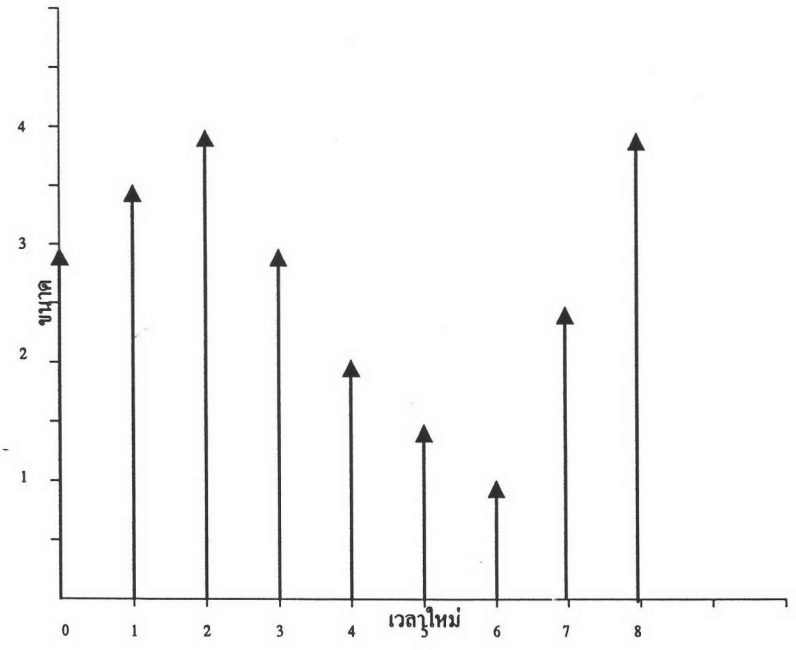
ตัวอย่างของการนอร์แมลไลซ์สัญญาณเสียงพูด แสดงในรูปที่ 3.6 ซึ่งแสดงการนอร์แมลไลซ์สัญญาณขนาด 5 จุดข้อมูลในรูปที่ 3.6(ก) เป็นสัญญาณขนาด 9 จุดข้อมูลในรูปที่ 3.6(ค) ในขั้นแรกคำนวณค่า timestep ได้เท่ากับ 0.5 สัญญาณนอร์แมลไลซ์จุดแรกมีค่าเท่ากับสัญญาณเสียงพูดเดิมจุดแรก สัญญาณนอร์แมลไลซ์จุดที่สองซึ่งปรากฏบนแกนเวลาใหม่ที่เวลา 1 มีค่าเท่ากับสัญญาณเสียงพูดเดิมบนแกนเวลาเดิมที่เวลาเท่ากับ timestep เนื่องจากสัญญาณเสียงพูดเดิมเป็นสัญญาณไม่ต่อเนื่อง ทำให้ไม่ทราบค่าสัญญาณที่แท้จริงที่เวลาไม่เท่ากับจำนวนเต็ม จึงทำการประมาณค่าในช่วงเชิงเส้น (linear interpolation) จากสัญญาณเสียงพูดเดิม 2 จุด ที่อยู่ใกล้ที่สุดที่เวลาเป็นจำนวนเต็ม คือสัญญาณเสียงพูดเดิมที่เวลา 0 และที่เวลา 1 ดังแสดงในรูปที่ 3.6(ข) สัญญาณนอร์แมลไลซ์ที่เวลาอื่นคำนวณจากวิธีเดียวกันนี้



(ก) สัญญาณเสียงพูดเดิม



(ข) การประมาณค่าในช่วงเชิงเส้นของสัญญาณนอร์แมลไลซ์จุดที่สอง

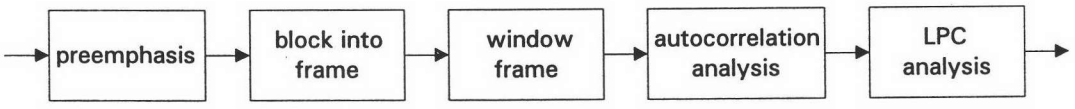


(ค) สัญญาณนอร์แมลไลซ์

รูปที่ 3.6 ตัวอย่างของการนอร์แมลไลซ์สัญญาณเสียงพูด

3.3 การประมาณพันธะเชิงเส้น

เป็นการวิเคราะห์เพื่อค่าพารามิเตอร์ที่เหมาะสมให้กับโครงสร้างของระบบกำเนิดเสียง โดยแบ่งสัญญาณเสียงพูดที่จะวิเคราะห์ออกเป็นส่วนย่อย แต่ละส่วนมีความยาวเป็นเวลาประมาณ 15-50 มิลลิวินาที (ms) (Rabiner and Levinson,1981) โดยที่ภายในเวลาช่วงสั้น ๆ นี้ สัญญาณเสียงพูดจะมีการเปลี่ยนแปลงคุณลักษณะอย่างช้า ๆ จนอาจถือว่าระบบกำเนิดเสียงมีคุณลักษณะไม่เปลี่ยนแปลง (stationary) การประมาณพันธะเชิงเส้นมีขั้นตอนการวิเคราะห์ดังรูปที่ 3.7 (Rabiner and Levinson,1981) โดยที่สัญญาณที่ป้อนเข้าเป็นสัญญาณเสียงพูดที่ผ่านการนอร์แมลไลซ์แล้ว



รูปที่ 3.7 ขั้นตอนการคำนวณค่าสัมประสิทธิ์ LPC

3.3.1 การเน้นล่วงหน้า (Preemphasis) เป็นการอัดพิสัยพลวัตของสัญญาณ (signal dynamic range) ทำให้อัตราส่วนสัญญาณต่อสัญญาณรบกวนมีค่าสูงขึ้น โดยใช้วงจรกรองดิจิตอลอันดับหนึ่ง

$$H(z) = 1 - az^{-1} \quad ; a = 0.95 \tag{3.3}$$

$$\tilde{s}(n) = s(n) - as(n-1) \tag{3.4}$$

3.3.2 การแบ่งสัญญาณเสียงพูดเป็นส่วนย่อย จะแบ่งให้แต่ละส่วนมีความยาวเป็นเวลาประมาณ 15-50 ms

$$x_l(n) = \tilde{s}(Ml + n) \qquad ; n = 0,1,\ldots,N-1$$
$$l = 0,1,\ldots,L-1 \tag{3.5}$$

เมื่อ N เป็นจำนวนจุดข้อมูลใน 1 ส่วนย่อย คำนวณได้จากความถี่ในการสุ่มสัญญาณคูณกับช่วงเวลาในส่วนย่อย

L เป็นจำนวนส่วนย่อยในสัญญาณเสียงพูด 1 คำ

M เป็นจำนวนจุดข้อมูลใน 1 ส่วนย่อยที่ไม่เหลื่อมกับข้อมูลในส่วนย่อยอื่น

สัญญาณเสียงพูดที่แบ่งเป็นส่วนย่อย จะมีการเหลื่อมกับสัญญาณเสียงพูดส่วนย่อยที่อยู่ติดกัน เพื่อให้ค่าสัมประสิทธิ์ LPC ของแต่ละส่วนย่อยมีความต่อเนื่องกัน โดยทั่วไปการเหลื่อมของสัญญาณเสียงพูดส่วนย่อยมีค่าระหว่าง 0 ถึง 1/2 ส่วนย่อย

3.3.3 การวางกรอบขนาดสัญญาณ (Window) เป็นการนำสัญญาณเสียงพูดในแต่ละส่วนย่อยมาผ่านฟังก์ชันที่กำหนด โดยในที่นี้ใช้ Hamming window (Furui,1989) ซึ่งจะค่อย ๆ ลดทอนแอมพลิจูดที่ปลายทั้งสองด้านของส่วนย่อย เพื่อป้องกันการเปลี่ยนแปลงอย่างรวดเร็วที่จุดปลาย

$$\tilde{x}_l(n) = x_l(n).w(n) \tag{3.6}$$

$$w(n) = 0.54 - 0.46\cos(2\pi n / N - 1) \tag{3.7}$$

3.3.4 การวิเคราะห์ค่าอัตสหสัมพันธ์ (Autocorrelation Analysis)

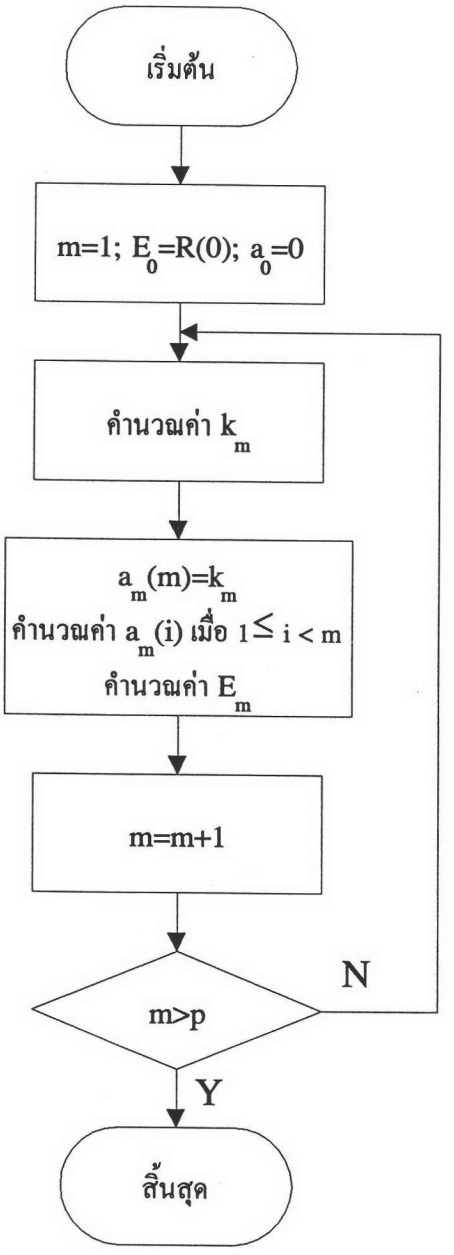
$$R_l(m) = \sum_{n=0}^{N-1-|m|} \tilde{x}_l(n)\tilde{x}_l(n+m) \quad ; m = 0,1,\ldots,p \tag{3.8}$$

เมื่อ p เป็นอันดับ (order) ของการวิเคราะห์ระบบ โดยทั่วไปค่าอันดับที่ใช้ในการวิเคราะห์มีค่าอยู่ระหว่าง 8-12 (Rabiner and Levinson,1981) เงื่อนไขในการเลือกค่าอันดับสำหรับการประมาณพันธะเชิงเส้นเป็นการปรับแต่งระหว่าง ความถูกต้องของสเปกตรัม,เวลาในการคำนวณและหน่วยความจำที่ต้องการ ถ้าใช้อันดับในการวิเคราะห์สูงจะทำให้สามารถประมาณสเปกตรัมได้อย่างแม่นยำ แต่จะทำให้สิ้นเปลืองหน่วยความจำและเวลาในการคำนวณ

$$X(l) = \{R_l(0), R_l(1), \ldots, R_l(p)\}$$

$$= \{R(0), R(1), \ldots, R(p)\} \text{ สำหรับสัญญาณเสียงพูดส่วนย่อยที่} l \tag{3.9}$$

3.3.5 การวิเคราะห์ค่าสัมประสิทธิ์ LPC เป็นการหาค่าสัมประสิทธิ์ LPC โดยใช้วิธีของ Levinson-Durbin ที่กล่าวไว้ในหัวข้อ 2.1.2 ขั้นตอนของการคำนวณค่าสัมประสิทธิ์ LPC แสดงในรูปที่ 3.8 โดยค่าต่าง ๆ คำนวณได้จากสมการ 2.9



รูป 3.8 ขั้นตอนการหาค่าสัมประสิทธิ์ LPC โดยใช้วิธีของ Levinson-Durbin

การคำนวณค่าสัมประสิทธิ์ LPC จะคำนวณจากค่าอัตสหสัมพันธ์ (autocorrelation) ของแต่ละส่วนย่อย โดยเมื่อทำการคำนวณตามขั้นตอนในรูปที่ 3.8 แล้วจะได้ค่าสัมประสิทธิ์ $a_m(1), a_m(2), \ldots, a_m(p)$ ซึ่งเป็นการประมาณพันธะเชิงเส้นที่เหมาะที่สุดสำหรับสัญญาณเสียงพูดส่วนย่อยนั้น ดังนั้นสัญญาณเสียงพูดแต่ละคำจะถูกแทนด้วย ชุดของสัมประสิทธิ์ LPC อันดับ p จำนวน L ชุด ชุดของสัมประสิทธิ์ LPC เหล่านี้จะถูกใช้เป็นข้อมูลอินพุตสำหรับนิวรอลเน็ตเวิร์กเพื่อใช้ในการฝึกหรือใช้ในการทดสอบ

3.4 นิวรอลเน็ตเวิร์ก

Multi-layer perceptron neural network เป็นนิวรอลเน็ตเวิร์ก ประเภทที่ต้องมีการเรียนรู้ของระบบ (supervised learning) ก่อนที่ระบบจะสามารถใช้ในการรู้จำได้ นิวรอลเน็ตเวิร์กมีจำนวน 2 ชุด โดยที่นิวรอลเน็ตเวิร์กชุดแรกใช้ในการรู้จำเสียงพูดพยางค์เดียว และนิวรอลเน็ตเวิร์กชุดที่สองใช้ในการรู้จำเสียงพูด 2 และ 3 พยางค์

3.4.1 โครงสร้างของนิวรอลเน็ตเวิร์ก

นิวรอลเน็ตเวิร์กที่ใช้ในวิทยานิพนธ์นี้ประกอบด้วย ระดับข้อมูลเข้า (input layer) ซึ่งใช้เป็นที่เก็บค่าอินพุตที่ป้อนให้กับเน็ตเวิร์ก, ระดับซ่อนตัว (hidden layer) และระดับข้อมูลออก (output layer) ซึ่งใช้สำหรับแสดงค่าเอาต์พุตของเน็ตเวิร์ก โดยที่ระดับซ่อนตัว (hidden layer) มีเพียง 1 ระดับ เพราะนิวรอลเน็ตเวิร์ก 3 ระดับสามารถประมาณฟังก์ชันต่อเนื่องใด ๆ ได้ (Schalkoff,1992), (KUNG,1993)

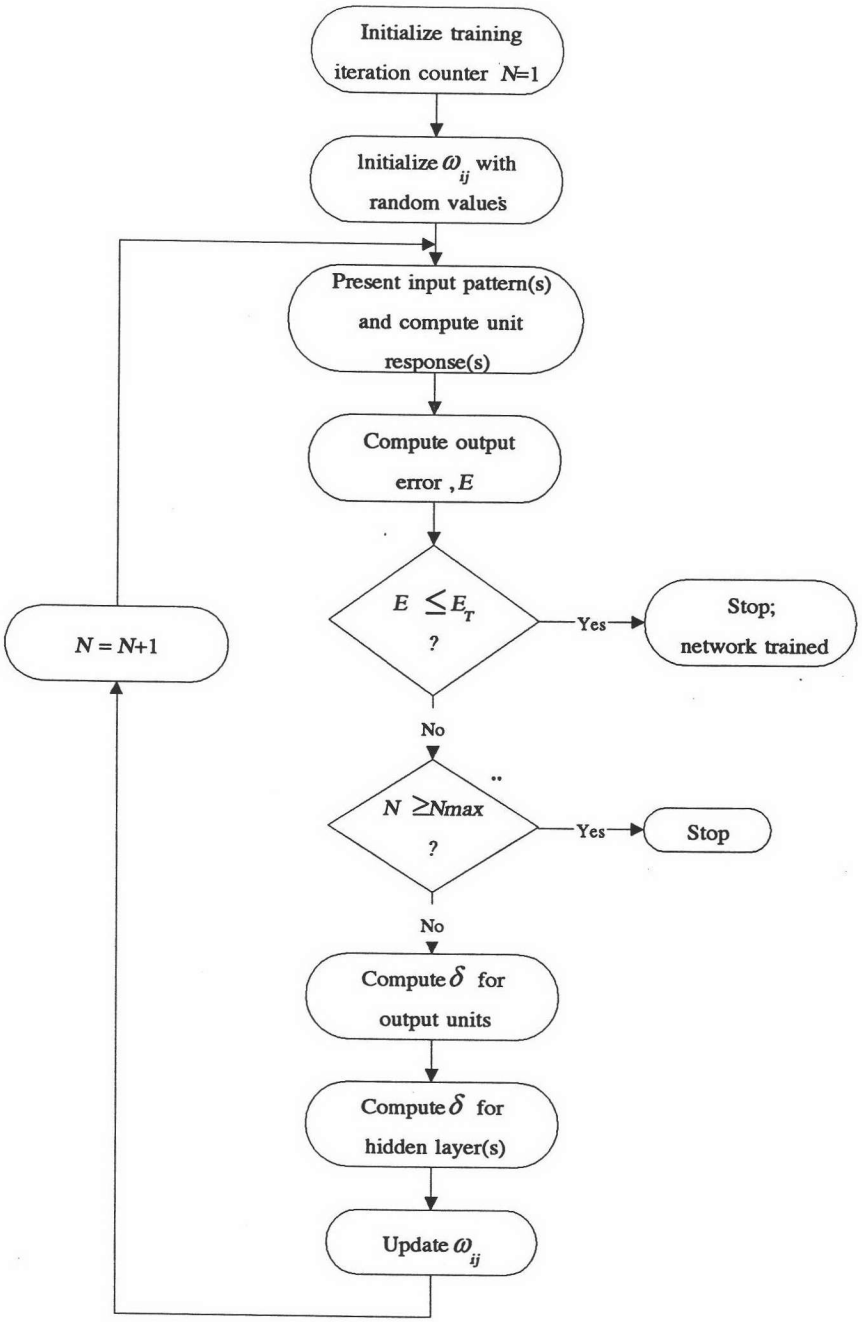
เงื่อนไขในการเลือกจำนวนโหนดในระดับต่าง ๆ มีดังต่อไปนี้ จำนวนโหนดในระดับข้อมูลเข้า (input layer) กำหนดโดยขนาดของข้อมูลอินพุตที่ใช้แทนเสียงพูด 1 คำ คือ L ส่วนย่อยและ 1 ส่วนย่อยประกอบด้วยสัมประสิทธิ์ LPC p ค่า ดังนั้นจำนวนโหนดในระดับข้อมูลเข้ามีค่าเท่ากับ pL โหนด จำนวนโหนดในระดับข้อมูลออก (output layer) กำหนดโดยจำนวนกลุ่มข้อมูลที่ต้องการแบ่งแยก ในที่นี้คือจำนวนเสียงพูดที่ต้องการรู้จำ สำหรับจำนวนโหนดในระดับซ่อนตัว (hidden layer) ซึ่งเป็นตัวกำหนดความยืดหยุ่นของขอบเขตการตัดสินใจ จะต้องทดลองเพื่อหาค่าที่เหมาะสม เพราะถ้ากำหนดให้มีจำนวนโหนดมากก็จะเสียเวลาในการฝึกนาน และอาจทำให้นิวรอลเน็ตเวิร์กจำลักษณะเฉพาะของเสียงในข้อมูลฝึกมากเกินไป ถ้ากำหนดให้มีจำนวนโหนดน้อยเกินไป นิวรอลเน็ตเวิร์กอาจไม่มีความสามารถพอสำหรับการจำลักษณะทั่ว ๆ ไปของเสียงในข้อมูลฝึก ดังนั้นจะทดลองฝึกนิวรอลเน็ตเวิร์กโดยใช้จำนวนโหนดในระดับซ่อนตัว (hidden layer) มีค่าหลาย ๆ ค่า แล้วค่อย ๆ ลดจำนวนโหนดลงเพื่อหาจำนวนโหนดที่น้อยที่สุดที่นิวรอลเน็ตเวิร์กยังมีความสามารถในการจำลักษณะทั่ว ๆ ไปของเสียงได้

3.4.2 การทำงานของนิวรอลเน็ตเวิร์ก

การทำงานของนิวรอลเน็ตเวิร์กในขณะฝึก (training) สามารถแบ่งเป็น 2 ลักษณะคือ การแพร่กระจายแบบไปข้างหน้า (feed forward) และการแพร่กระจายในทิศย้อนกลับ (back propagation) โดยในส่วนแรกจะเป็นการป้อนชุดของสัมประสิทธิ์ LPC เป็นข้อมูลอินพุตให้กับนิวรอลเน็ตเวิร์ก จากนั้นนิวรอลเน็ตเวิร์กจะทำการคำนวณจากระดับข้อมูลเข้า (input layer) ไปยังระดับข้อมูลออก (output layer) เพื่อหาค่าเอาต์พุตของทุกโหนด และในส่วนที่สองเป็นการคำนวณ

ในทิศย้อนกลับจากเอาต์พุตมายังอินพุต โดยเป็นการหาค่าความผิดพลาดระหว่างค่าเอาต์พุตที่ได้กับค่าเอาต์พุตที่ต้องการ และลดค่าความผิดพลาดโดยการปรับค่าน้ำหนักการเชื่อมต่อ (connection weight) ที่เชื่อมต่อระหว่างโหนดแต่ละระดับ (layer) การทำงานทั้งสองลักษณะจะถูกทำซ้ำไปเรื่อย ๆ โดยการใช้ตัวอย่างคู่ข้อมูลอินพุตเอาต์พุตในชุดฝึก จนกว่าจะได้ผลตามที่ต้องการ ส่วนการทำงานของนิวรอลเน็ตเวิร์กในขณะทดสอบ จะมีแต่การแพร่กระจายแบบไปข้างหน้า เพื่อหาค่าเอาต์พุตของทุกโหนดเท่านั้นและใช้กฎเกณฑ์ในการตัดสินใจ เพื่อเลือกว่าเสียงพูดเป็นเสียงใด

กระบวนการเรียนรู้แบบ backpropagation ที่ได้กล่าวไปแล้วนั้น แสดงในรูปที่ 3.9 โดยที่ค่าต่าง ๆ สามารถคำนวณได้จากสมการในหัวข้อที่ 2.4



รูปที่ 3.9 ขั้นตอนกระบวนการการเรียนรูปแบบ backpropagation

โดย N แทนจำนวนรอบในการเรียนรู้

N_{max} แทนจำนวนรอบสูงสุดที่ใช้ในการเรียนรู้

E แทนค่า output error

E_T แทนค่าระดับ output error ที่ต้องการเมื่อ E น้อยกว่าค่านี้ให้ยุติการฝึก

δ แทนค่าความไว (sensitivity) ของ pattern error เทียบกับ net activation

3.4.3 ตัวแปรที่สำคัญและเงื่อนไขในการกำหนดค่า

ค่า E_T ในรูปที่ 3.9 เป็นค่าระดับความผิดพลาดที่ต้องการ ซึ่งจะใช้เป็นเงื่อนไขในการหยุดการฝึก (training) ถ้าค่า E_T มีค่ามากเกินไปนิวรอลเน็ตเวิร์กจะไม่สามารถแบ่งแยกข้อมูลแต่ละกลุ่มได้ ถ้าค่า E_T มีค่าน้อยมากนิวรอลเน็ตเวิร์กอาจมีการจำลักษณะเฉพาะของเสียงในข้อมูลฝึกมากเกินไปและจะใช้เวลาในการฝึกนาน ตัวแปรสำคัญในการฝึก (training) นิวรอลเน็ตเวิร์กคือ learning rate และ momentum

ค่า learning rate กำหนดสัดส่วนในการปรับค่าน้ำหนักการเชื่อมต่อ โดยทั่วไปเป็นค่าสุ่มมีค่าตั้งแต่ 0 ถึง 1 การเลือกค่า learning rate ที่เหมาะสมขึ้นกับคุณลักษณะของพื้นผิวความผิดพลาด (error surface) ถ้าพื้นผิวมีการเปลี่ยนแปลงอย่างรวดเร็ว ควรเลือกค่า learning rate ให้มีค่าน้อย ๆ ถ้าพื้นผิวค่อนข้างราบเรียบควรเลือกค่า learning rate มากขึ้น เพื่อลดเวลาที่ใช้ในการฝึก (training) แต่ถ้ามีค่ามากเกินไปอาจทำให้เกิดการ oscillation และทำให้การลู่เข้าของนิวรอลเน็ตเวิร์กช้าหรือไม่สำเร็จ เนื่องจากคุณลักษณะของพื้นผิวความผิดพลาดเป็นสิ่งที่ไม่ทราบ ดังนั้นกฎเกณฑ์ในการเลือกค่า learning rate คือเลือกค่ามากที่สุดที่ใช้ได้ และไม่ทำให้เกิด oscillation

ค่า momentum เป็นการนำค่าการปรับค่าน้ำหนักการเชื่อมต่อในรอบก่อน มาใช้ในการคำนวณการปรับค่าน้ำหนักการเชื่อมต่อในรอบปัจจุบัน ค่า momentum อาจช่วยไม่ให้เกิด oscillation และป้องกันการติด local minimum ในขณะฝึก ค่า momentum เหมือนกับค่า learning rate ตรงที่ค่าที่เหมาะสมจะขึ้นกับคุณลักษณะของพื้นผิวความผิดพลาด การใช้ค่า momentum ทำให้สมการสำหรับการปรับค่าน้ำหนักการเชื่อมต่อในรอบที่ n+1 ถูกดัดแปลงเป็น

$$\Delta^p \omega_{ji}(n+1) = \varepsilon(1-\alpha)\delta_j^p \tilde{o}_i^p + \alpha \Delta^p \omega_{ji}(n) \qquad (3.10)$$

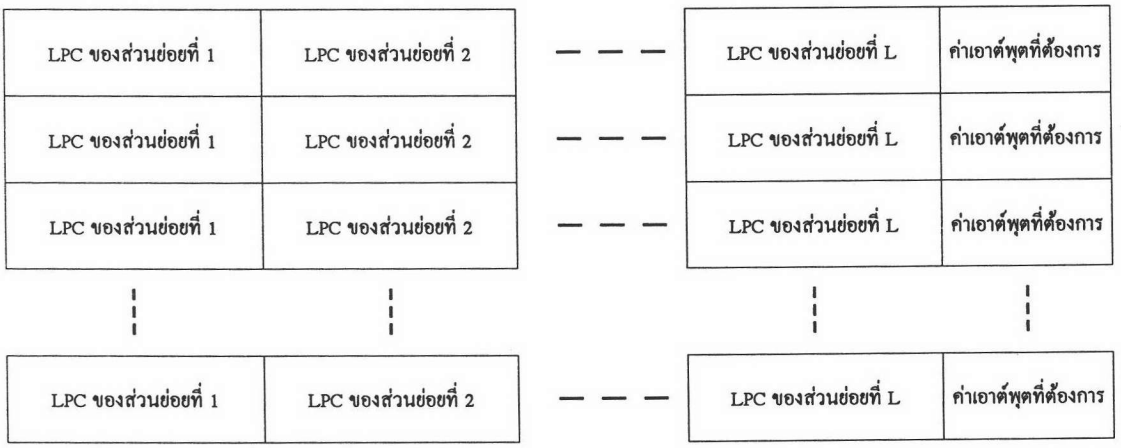
ขั้นตอนการให้ค่าเริ่มต้นแก่ค่าน้ำหนักการเชื่อมต่อด้วยการสุ่ม ทำเพื่อป้องกันการสมมาตร (symmetry) ของค่าน้ำหนักการเชื่อมต่อในนิวรอลเน็ตเวิร์ก (Schalkoff,1992) ถ้าหากนิวรอลเน็ตเวิร์กมีค่าเริ่มต้นของน้ำหนักการเชื่อมต่อทุกค่าเท่ากัน การปรับค่าน้ำหนักการเชื่อมต่อของแต่ละโหนดในระดับซ่อนตัว (hidden layer) จะมีค่าเท่ากัน ทำให้นิวรอลเน็ตเวิร์กทำตัวเสมือนมีโหนดในระดับซ่อนตัว (hidden layer) เพียงโหนดเดียว

3.4.4 การเก็บตัวอย่างคู่ข้อมูลอินพุตเอาต์พุต

เนื่องจากนิวรอลเน็ตเวิร์กต้องใช้ตัวอย่างข้อมูลในการฝึกจำนวนมาก ทำให้ต้องใช้หน่วยความจำจำนวนมากในการเก็บข้อมูล จึงแก้ปัญหานี้โดยการเก็บข้อมูลในแฟ้มข้อมูล เมื่อ

ต้องการใช้ตัวอย่างคู่ข้อมูลอินพุตเอาต์พุตคู่ใด ก็จะอ่านค่าจากแฟ้มข้อมูลเข้ามาสู่หน่วยความจำของระบบเพื่อป้อนค่าเข้าสู่นิวรอลเน็ตเวิร์ก

ข้อมูลที่ใช้ในการฝึกนิวรอลเน็ตเวิร์ก ประกอบด้วยคู่ของชุดสัมประสิทธิ์ LPC และค่าเอาต์พุตที่ต้องการของสัญญาณเสียงพูดแต่ละคำเรียงต่อกันในแฟ้มข้อมูล training.dat กล่าวคือ แฟ้มข้อมูลประกอบด้วยชุดสัมประสิทธิ์ LPC และค่าเอาต์พุตที่ต้องการของสัญญาณเสียงพูดคำแรก เรียงต่อด้วยชุดสัมประสิทธิ์ LPC และค่าเอาต์พุตที่ต้องการของสัญญาณเสียงพูดคำที่สองและคำที่สามต่อไปเรื่อย ๆ จนถึงชุดสัมประสิทธิ์ LPC และค่าเอาต์พุตที่ต้องการของสัญญาณเสียงพูดคำสุดท้าย ชุดสัมประสิทธิ์ LPC ที่คำนวณจากสัญญาณเสียงพูดของกลุ่มบุคคลที่เป็นเสียงต้นแบบทั้งหมดถูกบรรจุในแฟ้มข้อมูลนี้ รูปที่ 3.10 แสดงโครงสร้างของแฟ้มข้อมูลที่ใช้เก็บตัวอย่างคู่ข้อมูลอินพุตเอาต์พุตที่ใช้ในการฝึกนิวรอลเน็ตเวิร์ก โดยที่ค่าสัมประสิทธิ์ LPC และค่าเอาต์พุตที่ต้องการในแนวระดับเดียวกันเป็นตัวอย่างคู่ข้อมูลอินพุตเอาต์พุตของเสียงเดียวกัน



รูปที่ 3.10 โครงสร้างของแฟ้มข้อมูลที่ใช้เก็บตัวอย่างคู่ข้อมูลอินพุตเอาต์พุต

3.4 กฎเกณฑ์การตัดสินใจ

ดังที่ได้กล่าวไปแล้วในหัวข้อ 2.5 ว่ากฎเกณฑ์การตัดสินใจที่เลือกใช้คือ เลือกเสียงที่ตรงกับโหนดเอาต์พุตที่มีค่าเอาต์พุตสูงสุด ซึ่งโหนดเอาต์พุตที่มีค่าเอาต์พุตสูงสุดคำนวณได้จากสมการ 2.18 จากนั้นนำค่าโหนดเอาต์พุตที่เลือกไปเปิดตารางเพื่อดูว่าโหนดเอาต์พุตที่เลือกตรงกับเสียงคำว่าอะไรในจำนวนเสียงพูดที่ต้องการรู้จำ c คำ