บทที่ 2
หลักการและทฤษฎีสำคัญ

2.1 การรู้จำคำพูด (Speech Recognition)

ดังที่ได้กล่าวมาแล้วว่า คำพูด (Speech) เป็นข้อมูลพื้นฐานที่น่าจะนำมาใช้ในการสื่อสารมากที่สุด ทั้งยังสามารถประยุกต์ระบบที่ใช้ข้อมูลเสียงเป็นหลัก กับงานสร้างสรรค์ต่างๆ ได้อีกมาก ดังนั้นจึงมีงานวิจัยมากมาย ที่พัฒนากระบวนการรู้จำคำพูด (Speech Recognition) ซึ่งถือเป็นหัวใจสำคัญสำหรับเครื่องจักรกลในการติดต่อสื่อสารกับมนุษย์

2.1.1 ปัญหาในการรู้จำคำพูด

1) ปัญหาที่เกิดจากการเชื่อมต่อกันของคำพูด (Coarticulation) และการลดทอนของหน่วยเสียง (Reduction) ซึ่งอาจเกิดจากความต่อเนื่องในการพูด

2) ความยากในการแยก (Segmentation) เสียงเป็นหน่วยย่อย เช่น คำ หรือพยางค์

3) ความหลากหลายของผู้พูด และอุปกรณ์ที่ใช้ในระบบ

4) มีฐานข้อมูลทางภาษา (Linguistic Knowledge) ไม่เพียงพอ

2.1.2 ชนิดของการรู้จำคำพูด

การรู้จำคำพูดมีหลายชนิด ขึ้นอยู่กับประเด็นที่ใช้ในการแยก สรุปได้ดังตารางที่ 2.1 การรู้จำโดยใช้คำเป็นตัวทาบเทียบ (Word Based Recognition) มีข้อดีคือ สามารถหลีกเลี่ยงปัญหาความต่อเนื่องระหว่างหน่วยเสียงได้ เหมาะสำหรับการรู้จำคำศัพท์จำนวนไม่มาก (ไม่เกิน 1,000 คำศัพท์) เนื่องจากมีการคำนวณมาก เมื่อเทียบกับการรู้จำโดยใช้หน่วยเสียงเป็นตัวทาบเทียบ (Phoneme Based Recognition) อย่างไรก็ตามวิธีการใช้คำเป็นตัวทาบเทียบนี้ จะไม่สามารถหลีกเลี่ยงปัญหาความต่อเนื่องระหว่างคำ ในการรู้จำคำพูดต่อเนื่อง (Continuous Speech) ได้ การรู้จำโดยใช้หน่วยเสียงเป็นตัวทาบเทียบ จำเป็นจะต้องมีส่วนอ้างอิงในการผสมหน่วยเสียงโดยขึ้นกับบริบท (Context-dependent Multiple Phoneme Reference) เพื่อตรวจสอบว่า การผสมของหน่วยเสียงที่ได้จากการรู้จำ ตรงกับเสียงคำใด วิธีการนี้ สามารถเพิ่มจำนวนคำศัพท์ที่จะจดจำได้มหาศาล โดยเพิ่มเติมข้อมูลในส่วนการผสมหน่วยเสียงดังกล่าว ปัญหาใหญ่ในการใช้หน่วยเสียงเป็นตัวทาบเทียบคือ ปัญหาความต่อเนื่องระหว่างหน่วยเสียง ทำให้ยากต่อการตัด และยากต่อการรู้จำ การแก้ปัญหาวิธีหนึ่งคือการใช้หน่วยที่ใหญ่กว่าหน่วยเสียงขึ้นมา แต่อาจจะเล็กกว่าคำ เป็นตัวทาบเทียบ เช่น หน่วยเสียงคู่

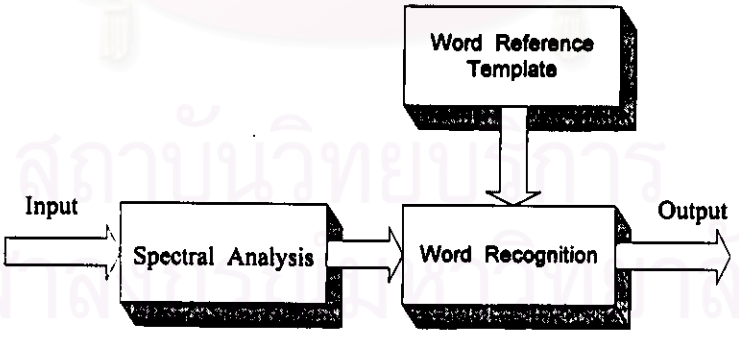
หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

(Diphone) เสียงสามหน่วย (Triphone) หรือพยางค์ที่เกิดจากพยัญชนะรวมกับสระ (CV Syllable) เป็นต้น

ตารางที่ 2.1 การแบ่งแยกชนิดของการรู้จำคำพูด

ประเด็นที่ใช้แบ่งแยก	ชนิดของการรู้จำ
การขึ้นกับผู้พูด	1. ขึ้นกับผู้พูด (Speaker Dependent) 2. ไม่ขึ้นกับผู้พูด (Speaker Independent)
ระดับของเสียงที่รู้จำ	1. การรู้จำคำเดี่ยว (Isolated Word) 2. การรู้จำคำพูดต่อเนื่อง (Continuous Speech) - คำต่อเนื่อง (Connected Word) - พูดเป็นประโยค (Conversational Speech)
หน่วยของเสียงที่ใช้เป็นตัวทาบเทียบ (Reference Template)	1. ใช้คำเป็นตัวทาบเทียบ (Word Based) 2. ใช้หน่วยเสียงเป็นตัวทาบเทียบ (Phoneme Based) 3. ใช้หน่วยที่อยู่ระหว่างคำ กับหน่วยเสียง เช่น หน่วยเสียงคู่ หรือเสียงสามหน่วย เป็นตัวทาบเทียบ

2.1.3 โครงสร้างของระบบรู้จำ



รูปที่ 2.1 โครงสร้างของการรู้จำเสียงคำเดี่ยว โดยใช้คำเป็นตัวทาบเทียบ

เนื่องจากงานวิจัยชิ้นนี้ จัดอยู่ในการรู้จำเสียงคำเดี่ยว (Isolated Word Recognition) ดังนั้น จะขอกล่าวถึงเฉพาะโครงสร้างของการรู้จำชนิดนี้ การรู้จำเสียงคำเดี่ยวทำได้ 2 วิธีคือ ใช้คำเป็นตัว

ทาบเทียบ หรือใช้หน่วยเสียงเป็นตัวทาบเทียบ ในที่นี้จะกล่าวถึงเฉพาะรายละเอียดของการรู้จำเสียงคำเดี่ยว โดยใช้คำเป็นตัวทาบเทียบ ซึ่งเป็นวิธีที่ใช้ในงานวิจัยนี้ มีโครงสร้างแสดงดังรูปที่ 2.1

1) การวิเคราะห์สเปกตรัมของเสียง (Spectral Analysis) เป็นขั้นตอนที่จะทำการสกัดเอาลักษณะเด่น (Feature) ของสัญญาณเสียง เพื่อนำมาเปรียบเทียบกับชุดคำอ้างอิง (Word Reference Template) โดยส่วนใหญ่ ลักษณะเด่นของเสียงจะเป็นค่าเชิงสเปกตรัม (Spectral Feature) เนื่องจาก สัญญาณเสียงมักถูกมองว่าเกิดจากการรวมกันของสัญญาณประเภท Sinusoidal หลายๆ ความถี่ และการรู้จำเสียงของมนุษย์ มักอาศัยลักษณะเด่นทางด้านนี้อยู่เสมอ ตัวอย่างค่าลักษณะเด่นเชิงสเปกตรัมที่นิยม ได้แก่ ค่าเซปสตรัม (Cepstrum) ค่าอัตราการตัดแกนศูนย์ (Zero-crossing Rate) และค่าสัมประสิทธิ์การประมาณพันธะเชิงเส้น (Linear Prediction Coefficient) (LPC) เป็นต้น

2) ชุดคำอ้างอิง (Reference Template) ใช้สำหรับเป็นตัวทาบเทียบกับคำที่เข้ามาทดสอบเพื่อรู้จำ ชุดคำอ้างอิงนี้จะได้มาจากขั้นตอนการฝึกฝนระบบรู้จำ (Training หรือ Learning)

3) การรู้จำเสียงคำ (Word Recognition) เป็นขั้นตอนที่ทำการเทียบคำที่เข้ามาทดสอบการรู้จำ กับคำแต่ละคำในชุดคำอ้างอิง โดยปกติ จะหาค่าระยะห่าง (Distance) เพื่อใช้ในการพิจารณาความคล้าย โดยระยะห่างของคำที่เข้ามาทดสอบ กับคำในชุดคำอ้างอิงตัวใดต่ำที่สุด แสดงว่าคำที่เข้ามาทดสอบตรงกับคำนั้น แต่ในทางปฏิบัติ ปัญหาเรื่องความหลากหลายของเสียงคำ ความหลากหลายของเวลาที่ใช้พูดคำนั้นๆ (Time Duration) ทำให้การเทียบ เพื่อหาระยะห่างไม่ใช่เรื่องง่าย จึงได้มีการพัฒนาวิธีการรู้จำแบบอื่นๆ ขึ้นมา เช่น แทนที่จะใช้ค่าลักษณะเด่นที่ได้จากการสกัดช่วงแรก มาเทียบกันโดยตรง ก็เปลี่ยนเป็นใช้แบบจำลองของฮิดเดนมาร์คอฟ เป็นตัวแทนของคำแต่ละคำ หรือใช้นิวรอลเน็ตเวอร์กในการรู้จำโดยตรง ซึ่งจะกล่าวถึงรายละเอียดในหัวข้อต่อไป ทำให้ลดจำนวนการคำนวณ และเพิ่มประสิทธิภาพของการรู้จำได้

2.1.4 แนวทางในการพัฒนาระบบรู้จำเสียง

แนวทางในการพัฒนาระบบรู้จำเสียง ขึ้นอยู่กับว่าจะนำไปประยุกต์ใช้ในงานแบบใด โดยหลักการพื้นฐานแล้ว จะมีแนวทางในการพัฒนาดังนี้

1) เพิ่มอัตราการรู้จำ (Recognition Rate) ซึ่งทำได้โดย

- ปรับปรุงระบบรู้จำ เช่น ใช้แบบจำลองฮิดเดนมาร์คอฟ ร่วมกับนิวรอลเน็ตเวอร์ก หรือเปลี่ยนวิธีการฝึกฝน หรือโครงสร้างของนิวรอลเน็ตเวอร์ก
- ปรับปรุงค่าลักษณะเด่นที่ป้อนให้กับระบบรู้จำ เช่น ใช้ข้อมูลแบบฟัซซี

2) เพิ่มจำนวนคำศัพท์ที่จะรู้จำ โดยพัฒนาเป็นระบบที่ใช้หน่วยเสียงเป็นตัวทาบเทียบ หรือพัฒนาเป็นระบบที่สามารถปรับเพิ่มจำนวนคำศัพท์ได้

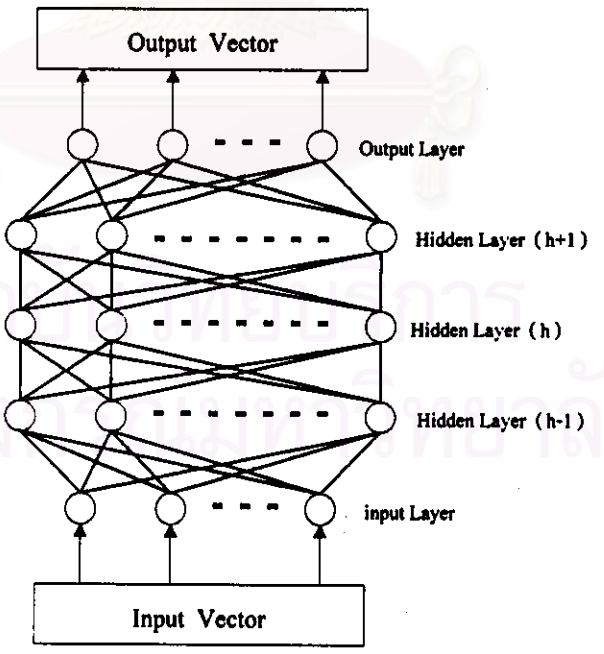
3) เพิ่มความเร็วของการรู้จำ (Recognition Time)

สำหรับในงานวิจัยนี้ ได้พัฒนาระบบรู้จำเสียงคำไทยหลายพยางค์แบบไม่ขึ้นกับผู้พูด ซึ่งจัดเป็นการรู้จำเสียงคำเดี่ยว โดยใช้คำเป็นตัวทาบเทียบ จุดสำคัญคือ ใช้ทฤษฎีทางฟัซซีมาปรับปรุงค่าลักษณะเด่น ที่สกัดได้จากสัญญาณเสียง และรู้จำโดยใช้นิวรอลเน็ตเวอร์ก ซึ่งจะมีหลักการดังจะกล่าวต่อไป

2.2 นิวรอลเน็ตเวอร์ก (Artificial Neural Network) (ANN)

นิวรอลเน็ตเวอร์กเป็นวิธีการหนึ่งที่นิยมนำมาใช้ในการรู้จำ มีพื้นฐานโครงสร้างเป็นการเลียนแบบโครงสร้างของสมองมนุษย์ ซึ่งมีลักษณะเป็นปมประสาทที่เชื่อมต่อกันแบบขนานกันจำนวนมาก ผลคือทำให้มีความสามารถในการคำนวณได้เร็วและจำนวนมากพร้อมกัน นับเป็นข้อดีประการสำคัญ นอกจากนี้ยังมีความสามารถในการแยกแยะรูปแบบที่จะรู้จำได้ดี และเร็วกว่าวิธีอื่นๆ เช่นวิธีแบบจำลองของฮิดเดนมาร์คอฟ (HMM)

นิวรอลเน็ตเวอร์กมีหลายแบบ ทั้งเป็น Supervised และ Unsupervised Learning ในงานวิจัยนี้มุ่งเน้นการรู้จำแบบ Supervised Learning นิวรอลเน็ตเวอร์กที่ใช้เป็นแบบ Multi-Layer Perceptron (MLP) โดยใช้วิธีการเรียนรู้แบบแพร่กระจายย้อนกลับ (Back-propagation Learning Algorithm) วิธีการนี้มีข้อดีคือเป็นวิธีพื้นฐานมีการคำนวณที่ไม่ซับซ้อน และสามารถรู้จำได้ดีไม่ด้อยกว่าวิธีอื่นๆ



รูปที่ 2.2 โครงสร้างของนิวรอลเน็ตเวอร์กแบบ MLP

2.2.1 โครงสร้างของนิวรอลเน็ตเวอร์กแบบ MLP

รูปที่ 2.2 แสดงโครงสร้างของนิวรอลเน็ตเวอร์กแบบ MLP ซึ่งประกอบด้วย ระดับชั้นข้อมูลเข้า (Input Layer) ระดับชั้นซ่อนตัว (Hidden Layer) ซึ่งอาจจะมีมากกว่า 1 ระดับ และระดับชั้นข้อมูลออก (Output Layer) ในแต่ละระดับชั้นจะประกอบด้วยโหนด (Node) หรือนิวรอน (Neuron) โดยในระดับชั้นข้อมูลเข้าจะมีจำนวนโหนดเท่ากับจำนวนข้อมูลในเวกเตอร์ข้อมูลเข้า (Input Vector) และในระดับชั้นข้อมูลออกมักจะมีจำนวนโหนดเท่ากับจำนวนรูปแบบที่จะทำการรู้จำ ที่แต่ละโหนดในระดับชั้นใดๆ จะมีค่าน้ำหนักการเชื่อมต่อ (Connection Weight) ไปยังโหนดที่อยู่ในระดับชั้นถัดไปเท่านั้น

2.2.2 ขั้นตอนในการฝึกฝน (Training หรือ Learning)

นิวรอลเน็ตเวอร์กแบบ MLP โดยอาศัยวิธีการฝึกฝนแบบ Back-propagation จะเรียนรู้จากแต่ละตัวอย่างของเวกเตอร์ข้อมูลเข้า ($\underline{i}$) และเวกเตอร์ข้อมูลออกที่ต้องการ ($\underline{t}$) ในชุดฝึกฝน (Training Set) มีขั้นตอนในการฝึกฝนดังนี้

1) ป้อนค่าเวกเตอร์ข้อมูลเข้า ให้กับระดับชั้นข้อมูลเข้า

2) แพร่กระจายไปข้างหน้า (Feed Forward) ค่าข้อมูลเข้าไปยังแต่ละระดับชั้นจนกระทั่งได้ค่าเวกเตอร์ข้อมูลออก รูปที่ 2.3 แสดงรายละเอียดของโหนด j ซึ่งอยู่ในระดับชั้นใดๆ (ระดับชั้นที่ h) ในนิวรอลเน็ตเวอร์กแบบ MLP ค่าข้อมูลออกของโหนด j (o_j) สามารถคำนวณได้จากสมการ

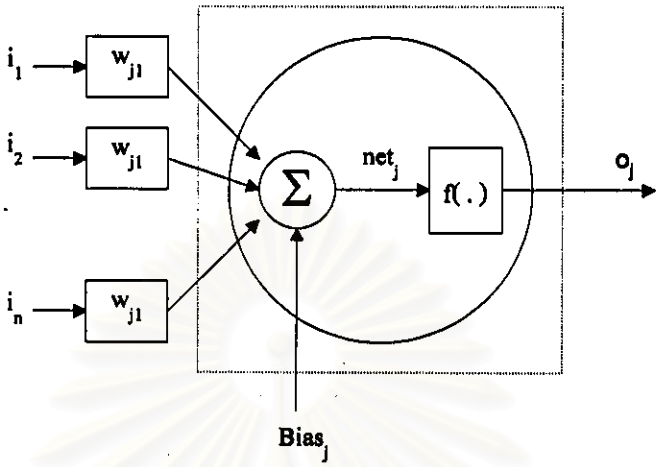
$$o_j = f(net_j) \tag{2.1}$$

โดยที่

$$net_j = \sum_i w_{ji} o_i + bias_j \tag{2.2}$$

เมื่อ o_i เป็นค่าของข้อมูลออกของโหนด i ซึ่งเป็นโหนดใดๆ ที่อยู่ในระดับชั้นก่อนหน้าโหนด j นี้ (ระดับชั้นที่ $h-1$) และจะเป็นค่าของข้อมูลเข้า i_i ถ้าโหนดที่ j เป็นโหนดในระดับชั้นข้อมูลเข้า w_{ji} เป็นค่าน้ำหนักการเชื่อมต่อจากโหนด i ไปยังโหนด j $bias_j$ เป็นค่าที่ใช้ปรับให้ net_j มีค่าไม่เป็นศูนย์ แม้ว่าค่า o_i จะมีค่าเป็นศูนย์หมด $f(net_j)$ เป็นค่าของฟังก์ชันกระตุ้นแบบไม่เป็นเชิงเส้น (Nonlinear Activation Function) ซึ่งต้องเป็นฟังก์ชันเพิ่ม และสามารถหาอนุพันธ์ได้ ในงานวิจัยนี้เลือกใช้ฟังก์ชัน Sigmoid ดังแสดงในสมการนี้

$$f(net_j) = \frac{1}{1+e^{-net_j}} \qquad (2.3)$$



รูปที่ 2.3 รายละเอียดของโหนดในนิวรอลเน็ตเวอร์กแบบ MLP

3) ปรับค่าน้ำหนักการเชื่อมต่อระหว่างโหนดใดๆ ด้วยวิธี Back-propagation โดยบวกเพิ่มค่า w_{ji} ด้วยค่า Δw_{ji} ซึ่งคำนวณได้จากสมการ

$$\Delta w_{ji}(n) = -\varepsilon \frac{\partial E}{\partial w_{ji}} + \alpha \Delta w_{ji}(n-1) \qquad (2.4)$$

โดยที่ ε คือค่า Learning Rate α คือค่า Momentum Rate ซึ่งเป็นค่าคงที่ทั้งคู่ พจน์ที่สองทางขวาของสมการที่ 2.4 เป็นพจน์ของโมเมนตัมซึ่งช่วยให้การปรับค่า w_{ji} ไม่เกิดการแกว่ง (Oscillation) มากเกินไป n เป็นตัวเลขแสดงลำดับครั้งของการปรับ โดยขณะเริ่มต้นจะตั้งค่า Δw_{ji} เป็นศูนย์ E เป็นความผิดพลาดของค่าข้อมูลออก เมื่อเทียบกับค่าข้อมูลออกที่ต้องการ ดังนี้

$$E = \frac{1}{2}\sum_j (t_j - o_j)^2 \qquad (2.5)$$

โดย o_j และ t_j เป็นค่าข้อมูลออก และค่าข้อมูลออกที่ต้องการของโหนดในระดับชั้นข้อมูลออก $j = 1, 2, ..., c$ โดยที่ c เป็นจำนวนโหนดในระดับชั้นข้อมูลออก ดังนั้นจะได้ว่า

$$\frac{\partial E}{\partial w_{ji}} = \frac{\partial E}{\partial o_j} \cdot \frac{do_j}{dnet_j} \cdot \frac{\partial net_j}{\partial w_{ji}} \qquad (2.6)$$

ถ้าโหนด j อยู่ในระดับชั้นข้อมูลออก จากสมการที่ 2.5 จะได้

$$\frac{\partial E}{\partial o_j} = \sum_j (t_j - o_j) \tag{2.7}$$

สำหรับโหนด j ที่อยู่ในระดับชั้นใดๆ (ระดับชั้นที่ h) นอกเหนือจากระดับชั้นข้อมูลเข้า

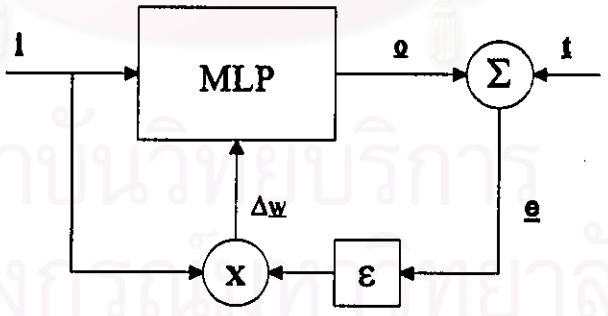
$$\frac{\partial E}{\partial o_j} = \sum_i \frac{\partial E}{\partial o_i} \frac{do_i}{net_i} \frac{\partial net_i}{\partial o_i} = \sum_i \frac{\partial E}{\partial o_i} \frac{do_i}{net_i} w_{ij} \tag{2.8}$$

ส่วน $\frac{do_j}{dnet_j}$ และ $\frac{\partial net_j}{\partial w_{ji}}$ ในสมการที่ 2.6 หาได้จาก

$$\frac{do_j}{dnet_j} = f'(net_j) = o_j(1 - o_j) \tag{2.9}$$

$$\frac{\partial net_j}{\partial w_{ji}} = o_i \tag{2.10}$$

โดยโหนด i อยู่ในระดับชั้นที่ถัดจากระดับชั้นของโหนด j ไปอีกหนึ่งชั้น (ระดับชั้นที่ $h+1$) จะเห็นได้ว่า การปรับค่าน้ำหนักการเชื่อมต่อ w_{ji} อาศัยการแพร่กระจายย้อนกลับ (Back-propagation) ของค่าความผิดพลาดของค่าข้อมูลออกของแต่ละโหนดในนิวรอลเน็ตเวอร์ก



รูปที่ 2.4 โครงสร้างขั้นตอนในการฝึกฝนแบบ Back-propagation

4) เมื่อปรับค่าน้ำหนักการเชื่อมต่อหมดทุกค่าแล้ว จะทำการแพร่กระจายค่าเวกเตอร์ข้อมูลเข้าอีกครั้งโดยใช้ค่าน้ำหนักการเชื่อมต่อค่าใหม่ที่ได้จากการปรับ กระทำเช่นนี้ต่อไปเรื่อยๆ จนกระทั่งค่าความผิดพลาด (E) ตามสมการที่ 2.7 น้อยกว่าค่าความผิดพลาดที่ยอมรับได้

(Error Threshold) เป็นอันว่าสิ้นสุดการฝึกฝนนิวรอลเน็ตเวอร์ก และจะเก็บค่าน้ำหนักการเชื่อมต่อสุดท้ายไว้สำหรับการทดสอบการรู้จำ สรุปขั้นตอนการฝึกฝนแบบ Back-propagation เป็นโครงสร้างดังรูปที่ 2.4

2.2.3 ขั้นตอนการรู้จำ (Recognition Test)

1) ป้อนค่าเวกเตอร์ข้อมูลของชุดทดสอบ (Test Set) เข้าไปยังระดับชั้นข้อมูลเข้า

2) แพร่กระจายค่าข้อมูลเข้า เข้าไปในนิวรอลเน็ตเวอร์กทำนองเดียวกับการฝึกฝน โดยอาศัยค่าน้ำหนักการเชื่อมต่อที่ได้จากการฝึกฝน จนกระทั่งได้ค่าเวกเตอร์ข้อมูลออกที่ระดับข้อมูลออก

3) ทำการตัดสินใจ โดยใช้กฎเกณฑ์การตัดสินใจ (Decision Rule) ดังนี้

เนื่องจากนิวรอลเน็ตเวอร์กแบบ MLP ที่ผ่านการฝึกฝนมาแล้ว จะให้ค่าข้อมูลออกที่คล้ายคลึงกับค่าข้อมูลออกที่ต้องการซึ่งคู่กับตัวอย่างข้อมูลเข้าในชุดฝึกฝน และโดยปกติจะกำหนดค่าข้อมูลออกที่ต้องการให้มีค่าสูงสุดที่โหนดที่สอดคล้องกับเวกเตอร์ข้อมูลเข้าที่ฝึกฝนขณะนั้น ดังนั้นเกณฑ์การตัดสินใจที่เลือกใช้ คือการเลือกรูปแบบ (Pattern) ที่ตรงกับโหนดในระดับชั้นข้อมูลออก ที่ให้ค่าข้อมูลออกมากที่สุด กฎเกณฑ์การตัดสินใจแบบนี้ เรียกว่าวิธี Maximum Likelihood (ML) ซึ่งเขียนเป็นสมการได้ดังนี้

$$\textit{Unknown Pattern} = \textit{Pattern}_i$$
$$\text{เมื่อ} \quad o_i = \max(o_1, o_2, \ldots, o_c) \qquad (2.11)$$

โดยที่ c คือจำนวนโหนดที่ระดับชั้นข้อมูลออก ซึ่งเท่ากับจำนวนคำศัพท์ที่จะรู้จำ

2.3 ทฤษฎีเกี่ยวกับฟัซซี (Fuzzy Theory)

ในงานทั่วๆไปที่ต้องมีการตัดสินใจ โดยที่ปัจจัยที่นำมาพิจารณาในการตัดสินใจมีความซับซ้อน กำกวม ยากต่อการตัดสินใจ จะพบว่ามนุษย์สามารถทำการตัดสินใจดีกว่าเครื่องจักรกล ขอยกตัวอย่างให้เห็นชัดเจน เช่น ต้องการจะคัดส้มเพื่อแบ่งแยกราคาขาย ปัจจัยที่ใช้ในการพิจารณาไม่ใช่เพียงขนาดของส้มเท่านั้น แต่ยังต้องพิจารณาสีผิวภายนอก น้ำหนักหรือความแน่นของเนื้อส้มและน้ำส้มในผลส้ม นอกจากนี้ยังอาจจะมีปัจจัยนอกเหนือจากที่คาดคิดออกไปได้อีก เช่น สภาวะเศรษฐกิจ สถานะทางบ้าน และอารมณ์ของเจ้าของผู้คัดส้ม เป็นต้น ในกรณีดังกล่าวนี้ เราไม่สามารถ

สร้างสมการทางคณิตศาสตร์ที่ครอบคลุมปัจจัยเหล่านี้เพื่อใช้ในการตัดสินใจได้ง่าย ทฤษฎีทางด้านฟัซซี (Fuzzy Theory) ถูกสร้างขึ้นมาเพื่อกรณีเหล่านี้

ข้อมูลที่พบอยู่ในโลกปัจจุบันสามารถแบ่งออกเป็นข้อมูลที่แน่นอน (Certain Information) และข้อมูลที่มีความไม่แน่นอน (Uncertain Information) ฟัซซีจัดเป็นทฤษฎีทางคณิตศาสตร์ที่ใช้กับข้อมูลที่มีความไม่แน่นอน เช่นเดียวกับทฤษฎีทางสถิติ (Statistic Theory) ต่อไปนี้จะกล่าวถึงทฤษฎีที่สำคัญเกี่ยวกับฟัซซี ซึ่งนำมาใช้ในงานวิจัยนี้

2.3.1 เซตแบบฟัซซี (Fuzzy Set)

เพื่อให้เข้าใจได้ง่าย จะขอกล่าวถึงเซตแบบแน่นอน (Crisp หรือ Classical Set) ก่อน โดยกำหนดให้ A และ B เป็นเซตที่ประกอบด้วยสมาชิก (Member) ดังนี้

$$\begin{aligned} A &= \{a,\ b,\ c\} \\ B &= \{b,\ c,\ d\} \end{aligned} \tag{2.12}$$

กำหนดให้ $\mu_X(y) \in [0,1]$ เป็นค่าสมาชิกภาพ (Membership Value) ของ y ในเซต X ซึ่งมีความหมายคือ $\mu_X(y)$ เป็นตัวเลขที่แสดงความเป็นสมาชิกของ y ใน เซต X ถ้า $\mu_X(y)$ มีค่ามาก แสดงว่า y มีความเป็นสมาชิกในเซต X มาก เมื่อพิจารณาเซตแบบแน่นอน A และ B จะพบว่า a, b และ c เป็นสมาชิกของเซต A แน่นอน และ b, c และ d เป็นสมาชิกของเซต B แน่นอน ดังนั้นค่าสมาชิกภาพจะเป็นดังนี้

$$\begin{aligned} \mu_A(a) &= 1 & \mu_B(a) &= 0 \\ \mu_A(b) &= 1 & \mu_B(b) &= 1 \\ \mu_A(c) &= 1 & \mu_B(c) &= 1 \\ \mu_A(d) &= 0 & \mu_B(d) &= 1 \end{aligned} \tag{2.13}$$

หรือเขียนรวมอยู่ในเซต โดยเขียนดังนี้

$$\begin{aligned} A &= \left\{\underset{1}{a},\ \underset{1}{b},\ \underset{1}{c},\ \underset{0}{d}\right\} \\ B &= \left\{\underset{0}{a},\ \underset{1}{b},\ \underset{1}{c},\ \underset{1}{d}\right\} \end{aligned} \tag{2.14}$$

ดังนั้น สำหรับเซตแบบแน่นอน ค่าสมาชิกภาพ $\mu_X(y)$ ของสมาชิกในเซตจะมีเพียง 1 หรือ 0 เท่านั้น สามารถเขียนสมการแสดงค่าสมาชิกภาพอยู่ในรูปทั่วไป ดังนี้

$$\mu_X(y)=\begin{cases}1, & y\in X\\ 0, & y\notin X\end{cases} \tag{2.15}$$

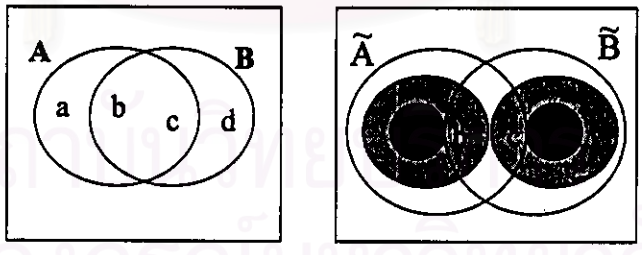
เซตแบบฟัซซี (Fuzzy Set) ประกอบด้วยสมาชิกที่มีค่าสมาชิกภาพ $\mu_X(y)\in[0,1]$ ยกตัวอย่างเช่น $\widetilde{A}$ และ $\widetilde{B}$ เป็นเซตแบบฟัซซีซึ่งมีสมาชิกและค่าสมาชิกภาพของสมาชิกแต่ละตัวดังนี้

$$\begin{aligned}\widetilde{A}&=\left\{\underset{1}{a},\ \underset{0.8}{b},\ \underset{0.5}{c},\ \underset{0}{d}\right\}\\ \widetilde{B}&=\left\{\underset{0}{a},\ \underset{0.4}{b},\ \underset{0.9}{c},\ \underset{1}{d}\right\}\end{aligned} \tag{2.16}$$

ซึ่งสามารถเขียนเซตแบบฟัซซีอยู่ในรูปทั่วไปได้ดังนี้

$$\widetilde{X}=\left\{\underset{\mu_{\widetilde{X}}(y)}{y}\right\},\quad \mu_{\widetilde{X}}(y)\in[0,1] \tag{2.17}$$

รูปที่ 2.5 เป็นแผนภาพของเซต A, B ซึ่งเป็นเซตแบบแน่นอน และเซต $\widetilde{A},\widetilde{B}$ ซึ่งเป็นเซตแบบฟัซซี



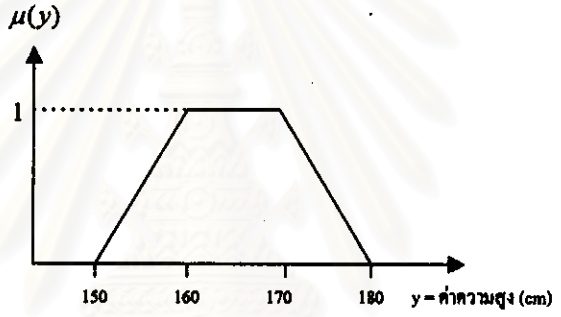
รูปที่ 2.5 แผนภาพแสดงสมาชิกในเซต A, B และ $\widetilde{A},\widetilde{B}$

2.3.2 ฟังก์ชันสมาชิกภาพแบบฟัซซี (Fuzzy Membership Function)

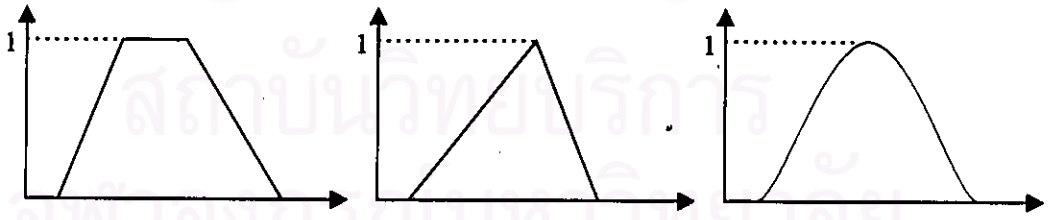
ฟังก์ชันสมาชิกภาพแบบฟัซซี $\mu(.)$ ก็คือฟังก์ชันที่ใช้หาค่าสมาชิกภาพ $\mu_{\widetilde{X}}(y)$ ของสมาชิก y ในเซตแบบฟัซซี $\widetilde{X}$ ในกรณีที่สมาชิกในเซตมีจำนวนจำกัด และไม่มากเกินไป สามารถเขียนแบบแจกแจงพร้อมทั้งแสดงค่าสมาชิกภาพได้ ดังเช่นในสมการที่ 2.16 แต่ถ้าสมาชิก

ในเซตเป็นค่าต่อเนื่อง เช่น เซต $\tilde{A}$ เป็นเซตของความสูงเป็นเซนติเมตรของคนที่จัดว่าเป็นคนสูงปานกลาง สมาชิกของเซตนี้อาจจะมีค่าสมาชิกภาพเป็นศูนย์ในช่วงความสูงน้อยกว่า 150 เซนติเมตร และความสูงมากกว่า 180 เซนติเมตร ส่วนในช่วงความสูงระหว่าง 150 ถึง 180 เซนติเมตร จัดอยู่ในเซตนี้ด้วยค่าสมาชิกภาพต่างๆ กัน นิยมแสดงฟังก์ชันสมาชิกภาพแบบฟัซซี เป็นกราฟความสัมพันธ์ของค่าสมาชิกภาพแบบฟัซซี เทียบกับค่าของสมาชิกทั้งหมดในเซต ดังเช่น กราฟในรูปที่ 2.6

ดังนั้น การกำหนดว่าฟังก์ชันสมาชิกภาพแบบฟัซซีจะเป็นกราฟรูปแบบใด ขึ้นอยู่กับลักษณะของข้อมูล และข้อกำหนดที่เราจะใส่ให้สำหรับข้อมูลชุดนั้น ซึ่งจะแตกต่างกันไปสำหรับข้อมูลในปัญหาต่างๆ กัน กราฟแสดงฟังก์ชันสมาชิกภาพแบบฟัซซีที่สำคัญ ได้แก่ ชนิดสี่เหลี่ยมคางหมู (Trapezoidal Function) ชนิดสามเหลี่ยม (Triangular Function) และชนิด Pi (π หรือ Pi Function) ดังรูปที่ 2.7



รูปที่ 2.6 ตัวอย่างกราฟความสัมพันธ์ของค่าสมาชิกภาพแบบฟัซซีกับค่าของสมาชิกในเซต $\tilde{A}$ ซึ่งเป็นเซตของความสูงของคนที่จัดว่าสูงปานกลาง



(ก) สี่เหลี่ยมคางหมู (ข) สามเหลี่ยม (ค) พาย

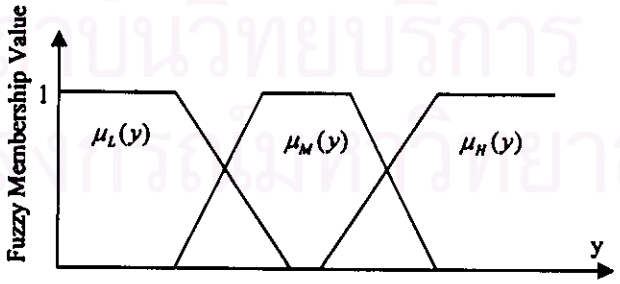
รูปที่ 2.7 กราฟแสดงฟังก์ชันสมาชิกภาพแบบฟัซซีทั้ง 3 ชนิด

สำหรับฟังก์ชันสมาชิกภาพชนิด Pi จะประกอบด้วยค่า c เป็นค่าของสมาชิกตัวที่มีค่าสมาชิกภาพเป็น 1 ซึ่งจะอยู่ตรงกึ่งกลางของรูปกราฟพอดี และ λ ซึ่งเป็นค่าที่ใช้กำหนดความกว้างของรูปกราฟ โดยเขียนเป็นสมการได้ดังนี้

$$\mu_{\tilde{X}}(y)=\begin{cases}2\left(1-\dfrac{|y-c|}{\lambda}\right)^2, & \dfrac{\lambda}{2}\le|y-c|\le\lambda\\ 1-2\left(\dfrac{|y-c|}{\lambda}\right)^2, & 0\le|y-c|\le\dfrac{\lambda}{2}\\ 0, & otherwise\end{cases} \qquad (2.18)$$

2.3.3 ฟังก์ชันสมาชิกภาพแบบฟัซซีบนคุณสมบัติทางภาษา (Linguistic Properties)

คุณสมบัติทางภาษา (Linguistic Properties) ในกรณีที่จะกล่าวนี้ มีความหมายแตกต่างจากความหมายทั่วไปเล็กน้อย จะขอยกตัวอย่างเพื่อให้เห็นภาพ ในกรณีของการตัดสินใจว่าคนๆ หนึ่งเป็นคนสูงมาก หรือน้อย โดยพิจารณาจากค่าความสูงของคนๆ นั้น คุณสมบัติทางภาษาในที่นี้อาจจะแบ่งออกเป็น 3 ระดับคือ { น้อย (L), ปานกลาง (M), มาก (H) } หรือ 5 ระดับคือ { น้อย (L), ค่อนข้างน้อย (SL), ปานกลาง (M), ค่อนข้างมาก (SH), มาก (H) } หลังจากนั้น เราจะกำหนดเซตแบบฟัซซี พร้อมทั้งฟังก์ชันสมาชิกภาพแบบฟัซซีสำหรับแต่ละสมาชิกในแต่ละเซต เช่น ในกรณีใช้คุณสมบัติทางภาษา 3 ระดับ เราอาจจะสร้างเซต $\tilde{A}_L$, $\tilde{A}_M$ และ $\tilde{A}_H$ เป็นเซตแบบฟัซซีของคนที่จัดว่าเตี้ย สูงปานกลาง และจัดว่าสูง ตามลำดับ โดยมีฟังก์ชันสมาชิกภาพแบบฟัซซีดังแสดงในกราฟรูปที่ 2.8



รูปที่ 2.8 ตัวอย่างกราฟแสดงฟังก์ชันสมาชิกภาพแบบฟัซซี บนคุณสมบัติทางภาษา 3 ระดับ

ดังนั้น เราสามารถแปลงค่าความสูงใดๆ (y) เป็นค่าสมาชิกภาพแบบฟัซซี (Fuzzy Membership Value) ได้เป็น 3 ค่า คือ $\mu_{\tilde{\lambda}_L}(y)$, $\mu_{\tilde{\lambda}_M}(y)$ และ $\mu_{\tilde{\lambda}_H}(y)$ ตามคุณสมบัติทางภาษา 3 ระดับ เรียกขั้นตอนนี้ว่าการแปลงเป็นค่าสมาชิกภาพแบบฟัซซี (Fuzzification) และค่าสมาชิกภาพแบบฟัซซีที่ได้ จะนำไปใช้ในกระบวนการต่อไป เช่น นำไปเป็นข้อมูลประกอบการตัดสินใจ ซึ่งให้ผลการตัดสินใจดีกว่าข้อมูลแบบปกติ

2.4 การใช้ทฤษฎีเกี่ยวกับฟัซซีร่วมกับนิวรอลเน็ตเวอร์กในการรู้จำ

ความผิดพลาดในการรู้จำรูปแบบ (Pattern Recognition) โดยเฉพาะการรู้จำคำพูดของมนุษย์ (Speech Recognition) มักเกิดจากความกำกวม ไม่แน่นอน ซึ่งเป็นธรรมชาติของคำพูด ทฤษฎีเกี่ยวกับฟัซซี ซึ่งสร้างขึ้น เพื่อให้สามารถรวมปัญหาเรื่องความกำกวมนี้ เข้าไปเป็นปัจจัยร่วมในการตัดสินใจ จึงถูกนำมาใช้ช่วยเพิ่มประสิทธิภาพในการรู้จำคำพูดอยู่เสมอ อย่างไรก็ตาม ดังตัวอย่างการใช้ทฤษฎีเกี่ยวกับฟัซซี ที่แสดงให้เห็นในหัวข้อที่ 2.3.3 จะเป็นการเพิ่มปริมาณการคำนวณขึ้นอย่างมาก นอกจากนี้ ค่าสมาชิกภาพแบบฟัซซีที่ได้อาจจะไม่สามารถนำมาใช้ในการตัดสินใจได้โดยตรง เนื่องจากเป็นข้อมูลที่พยายามรวมเรื่องความกำกวมมาเป็นปัจจัยในการตัดสินใจ จึงเป็นการเหมาะสมอย่างยิ่งที่จะใช้นิวรอลเน็ตเวอร์กแบบ MLP มาช่วยในการตัดสินใจ โดยอาศัยข้อมูลค่าสมาชิกภาพแบบฟัซซีเป็นข้อมูลเข้าในการฝึกฝน และการรู้จำ เนื่องจากนิวรอลเน็ตเวอร์กมีความสามารถในการคำนวณแบบขนานกันจำนวนมากๆ เป็นการช่วยลดระยะเวลาในการคำนวณ และยังมีความสามารถในการรู้จำแบบไม่เป็นเชิงเส้น (Nonlinear) ทำให้สามารถจดจำได้แม้ว่าข้อมูลเข้าจะมีความกำกวมอยู่บ้าง

มีงานวิจัยอยู่หลายชิ้นที่อาศัยทฤษฎีเกี่ยวกับฟัซซีร่วมกับนิวรอลเน็ตเวอร์กแบบ MLP ในการรู้จำคำพูด โดยมีหลักการพื้นฐานคล้ายกัน (Sankar and Sushmita, 1992) (Carlos and Wyllis, 1994) (Yingyong and Bobby, 1993) ดังที่จะกล่าวต่อไปนี้

2.3.1 เวกเตอร์ข้อมูลเข้า (Input Vector)

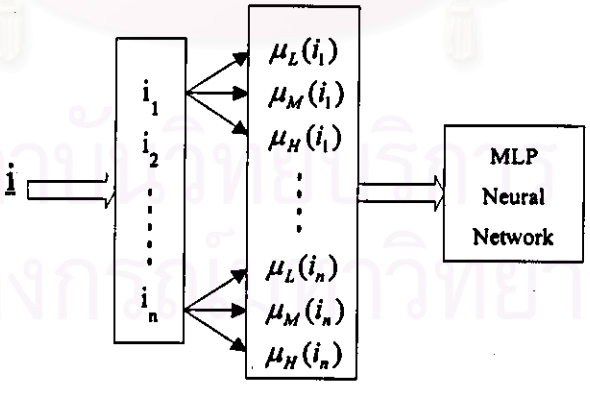
ปกติแล้ว ในการรู้จำคำพูดโดยใช้นิวรอลเน็ตเวอร์กแบบ MLP เราจะทำการสกัดลักษณะเด่น (Feature) บางประการของสัญญาณเสียงที่ผ่านกระบวนการประมวลผลเบื้องต้น (Preprocessing) มาสร้างเป็นเวกเตอร์ของลักษณะเด่น (Feature Vector) ดังนี้

$$\underline{i} = [i_1 \quad i_2 \quad i_3 \quad \ldots\ldots \quad i_n]^t \qquad (2.19)$$

โดยที่ i_j ใดๆ เป็นค่าลักษณะเด่น (Feature) ที่สกัดได้จากสัญญาณเสียง n เป็นจำนวนของลักษณะเด่นของสัญญาณเสียงหนึ่งๆ จะใช้เวกเตอร์นี้เป็นเวกเตอร์ข้อมูลเข้าของนิวรอลเน็ตเวอร์กแบบ MLP เพื่อทำการฝึกฝน และการรู้จำ แต่เมื่อสัญญาณเสียงมีความกำกวม อาจส่งผลให้การรู้จำผิดพลาด จึงทำการแปลงค่าลักษณะเด่นที่สกัดมานี้ เป็นค่าสมาชิกภาพแบบฟัซซี (Fuzzy Membership Value) บนคุณสมบัติทางภาษา (Linguistic Properties) ดังเช่นที่กล่าวมาในหัวข้อ 2.3.3 ถ้าใช้คุณสมบัติทางภาษา 3 ระดับคือ { ค่าน้อย (L), ค่าปานกลาง (M), ค่ามาก (H) } จะสามารถแปลงเวกเตอร์ของลักษณะเด่นในสมการที่ 2.19 ได้เป็น

$$\underline{\tilde{i}} = \begin{bmatrix} \mu_L(i_1) & \mu_M(i_1) & \mu_H(i_1) \\ \mu_L(i_2) & \mu_M(i_2) & \mu_H(i_2) \\ \cdots\cdots & & \\ \mu_L(i_n) & \mu_M(i_n) & \mu_H(i_n) \end{bmatrix}' \qquad (2.20)$$

โดย $\mu_L(i_j), \mu_M(i_j)$ และ $\mu_H(i_j)$ เป็นค่าสมาชิกภาพแบบฟัซซีของค่าลักษณะเด่น i_j ใดๆ ในเซตแบบฟัซซี L, M และ H ตามลำดับ สำหรับฟังก์ชันสมาชิกภาพแบบฟัซซีที่ใช้จะมีรูปร่างได้หลายแบบดังแสดงให้เห็นในหัวข้อ 2.3.2 และจะนำมาสร้างเป็นกราฟแสดงฟังก์ชันสมาชิกภาพแบบฟัซซี บนคุณสมบัติทางภาษา ให้ครอบคลุมปริภูมิของค่าลักษณะเด่นที่เกิดขึ้นได้ทั้งหมด (Feature Space) สรุปเป็นขั้นตอนการสกัดเวกเตอร์ข้อมูลเข้าเป็นค่าสมาชิกภาพแบบฟัซซีสำหรับเข้านิวรอลเน็ตเวอร์ก ได้ดังรูปที่ 2.9



รูปที่ 2.9 แผนภาพแสดงขั้นตอนการสกัดเวกเตอร์ข้อมูลเข้าเป็นค่าสมาชิกภาพแบบฟัซซี

2.4.2 เวกเตอร์ข้อมูลออกที่ต้องการ (Desired Output Vector)

สำหรับนิวรอลเน็ตเวอร์กแบบ MLP มักกำหนดให้มีจำนวนโหนดที่ระดับชั้นข้อมูลออก (Output Layer) เท่ากับจำนวนรูปแบบที่จะรู้จำ ในที่นี้ก็คือ จำนวนคำศัพท์ (c) ที่จะรู้จำคำพูด และเวกเตอร์ข้อมูลออกที่ต้องการ (Desired Output Vector) ก็จะมีมิติเท่ากับจำนวนโหนดนี้ สำหรับค่าในเวกเตอร์ข้อมูลออกแบบปกติ จะประกอบด้วยค่า 1 หรือ 0 (Binary) เท่านั้น โดยถ้าเวกเตอร์ข้อมูลเข้าสกัดมาจากเสียงของคำศัพท์ตัวที่ j ($j=1,2,...,c$) จะให้ค่าของข้อมูลออกที่ต้องการ (Desired Output Value) เท่ากับ 1 ที่โหนด j และมีค่าเป็น 0 ที่โหนดอื่นๆ ลักษณะของเวกเตอร์ข้อมูลออกที่ต้องการแบบนี้ ถือได้ว่าเป็นการฝึกฝนให้เกิดการตัดสินใจ โดยละเลยโอกาสที่ข้อมูลเข้าชุดนี้ จะไปคล้ายกับคำศัพท์ตัวอื่นๆ ซึ่งในชีวิตจริง การตัดสินใจของมนุษย์ว่าเสียงใดตรงกับคำศัพท์ตัวใด จะพิจารณาจุดนี้อยู่เสมอ ดังนั้นจึงมีการปรับปรุงค่าในเวกเตอร์ข้อมูลออกที่ต้องการ ในช่วงฝึกฝน เป็นค่าสมาชิกภาพของเวกเตอร์ข้อมูลเข้าต่อคำศัพท์แต่ละตัว (Class Membership Value) ซึ่งเป็นตัวเลขที่แสดงโอกาสที่เวกเตอร์ข้อมูลเข้าใดๆ จะตรงกับคำศัพท์แต่ละตัวที่ทำการรู้จำ

ค่าสมาชิกภาพของเวกเตอร์ข้อมูลเข้าต่อคำศัพท์แต่ละตัวนี้ จะนิยามได้จากค่าระยะห่าง (Distance) ของเวกเตอร์ข้อมูลเข้า จากคำศัพท์แต่ละตัว ดังนี้

$$z_k = \sqrt{\sum_{j=1}^{n}\left[\frac{i_j - o_{kj}}{v_{kj}}\right]^2} \qquad (2.21)$$

โดย z_k เป็นค่าระยะห่าง (Distance) จากคำศัพท์ตัวที่ k ($k=1,2,...,c$) ของเวกเตอร์ข้อมูลเข้า $\underline{i}$ ใดๆ ดังที่แสดงในสมการที่ 2.19 o_{kj} และ v_{kj} เป็นค่าเฉลี่ยเลขคณิต (Mean) และค่าความเบี่ยงเบน (Deviation) ของค่าลักษณะเด่นตัวที่ j ($j=1,2,...,c$) ในเวกเตอร์ข้อมูลเข้าทั้งหมดที่สกัดมาจากเสียงคำศัพท์ตัวที่ k ในชุดฝึกฝน (Training Set) ดังนั้น ถ้าเวกเตอร์ข้อมูลเข้าตัวหนึ่งๆ ที่สกัดมาจากเสียงคำศัพท์ตัวใด ย่อมมีค่าระยะห่าง z_k จากคำศัพท์ตัวนี้น้อยที่สุด และเราจะได้ค่าสมาชิกภาพของเวกเตอร์ข้อมูลเข้า $\underline{i}$ ใดๆ ต่อคำศัพท์ตัวที่ k ($\mu_k(\underline{i})$) ดังนี้

$$\mu_k(\underline{i}) = \frac{1}{1+\left(\frac{z_k}{f_d}\right)^{f_e}} \quad , \mu_k(\underline{i}) \in [0,1] \qquad (2.22)$$

โดย f_d และ f_e เป็นค่าที่ใช้ควบคุมความกำกวม (Fuzziness) สามารถปรับค่าได้ เพื่อกำหนดปรับค่า $\mu_k(i)$ ให้เหมาะสม จะเห็นว่าค่า $\mu_k(i)$ จะสวนทางกับค่าระยะห่าง z_k นั่นคือ ถ้าเวกเตอร์ข้อมูลเข้าสกัดมาจากเสียงคำศัพท์ตัวใด จะมีค่า $\mu_k(i)$ มากที่สุด เมื่อ k ตรงกับคำศัพท์ตัวนั้น และจะมีค่าน้อยเมื่อ k ตรงกับคำศัพท์ตัวอื่นๆ

ในทางปฏิบัติ ค่า $\mu_k(i)$ ใดๆ อาจมีค่ากำกวมคืออยู่ในช่วงกึ่งกลางระหว่าง 0 กับ 1 ในกรณีเช่นนี้ส่งผลให้การฝึกฝนนิวรอลเน็ตเวอร์กด้วยเวกเตอร์ข้อมูลออกที่ต้องการนี้ เกิดความผิดพลาดได้ เรียกลักษณะของข้อมูลออกที่ต้องการเช่นนี้ว่ากรณีกำกวมที่สุด (Fuzziest Case) งานวิจัยของ Pal (Pal and Mitra, 1992) ได้เสนอวิธีการในการแก้ไขกรณีดังกล่าว โดยปรับค่า $\mu_k(i)$ ที่ได้ใหม่ด้วยสมการปรับค่าฟัซซีชนิด INT ซึ่งมีสมการดังนี้

$$\mu_{INT(k)}(i) = \begin{cases} 2[\mu_k(i)]^2 & , 0 \le \mu_k(i) \le 0.5 \\ 1-2[1-\mu_k(i)]^2 & , otherwise \end{cases} \qquad (2.23)$$

ค่าข้อมูลออกที่ต้องการใหม่ $\mu_{INT(k)}(i)$ ที่ได้จากสมการที่ 2.23 จะมีค่าสูงมากขึ้น ถ้า $\mu_k(i)$ มีค่าสูงกว่า 0.5 เล็กน้อย และจะมีค่าต่ำลง ถ้า $\mu_k(i)$ มีค่าต่ำกว่า 0.5 เล็กน้อย เป็นการปรับให้ค่าข้อมูลออกที่ต้องการมีการแยกแยะกันเด่นชัดขึ้น

2.5 ทฤษฎีอื่นๆ ที่สำคัญ

2.5.1 การตัดหัวท้ายหน่วยที่ใช้ในการรู้จำ (Endpoint Detection) และตัดแบ่งพยางค์ (Syllable Segmentation) การตัดหัวท้ายหน่วยที่ใช้ในการรู้จำเป็นกระบวนการที่จะแยกส่วนที่เป็นคำพูด ออกจากส่วนที่ไม่ใช่คำพูด หรือส่วนที่เป็นเสียงพื้นหลัง (Background Sound) การตัวหัวท้ายคำถือว่าเป็นกระบวนการที่สำคัญ มีผลต่ออัตราการรู้จำค่อนข้างมาก (Pornsukchantra, 1996) และมีผลต่อเวลาในการคำนวณโดยตรง ส่วนการตัดแบ่งพยางค์จะทำการหาจุดแบ่งระหว่างพยางค์ในคำ มีประโยชน์ในการตรวจสอบจำนวนพยางค์ และการรู้จำเสียงโดยใช้พยางค์เป็นตัวทาบเทียบ

วิธีในการตัดหัวท้ายหน่วยที่ใช้ในการรู้จำจะรวมอยู่ในวิธีการในการตัดแบ่งพยางค์ มีวิธีหลักๆ ดังนี้

1) ใช้ค่าแอมปลิจูด

เริ่มต้นจาก หาค่าสัมบูรณ์ของแอมปลิจูดของสัญญาณเสียงแต่ละจุดที่ n ใดๆ ดังนี้

$$E(n) = \left| s(n) \right| \qquad (2.24)$$

และใช้หลักการตัดหัวท้ายหน่วยที่ใช้ในการรู้จำคือ เมื่อสัญญาณมีค่า $E(n)$ มากกว่าค่าที่กำหนด (Threshold) ติดกันมากกว่าจำนวนที่กำหนดไว้ (Duration Threshold) ถือได้ว่าเป็นช่วงเริ่มต้นของคำ และจะทำเช่นเดียวกันในส่วนท้ายของสัญญาณเสียง เพื่อหาจุดสิ้นสุดคำ วิธีการนี้อาจจะมีข้อผิดพลาดได้ ถ้ามีสัญญาณรบกวน (Noise) มากพอ จนทำให้เห็นเป็นคำได้ วิธีแก้ไขวิธีหนึ่งคือแทนที่จะพิจารณาค่าแอมปลิจูดทีละจุด ก็เปลี่ยนเป็น แบ่งสัญญาณเสียงเป็นส่วนย่อย (Frame) แล้วหาผลรวมของค่าสัมบูรณ์ของแอมปลิจูดของสัญญาณในส่วนย่อยที่ j ใดๆ ได้ดังนี้

$$E(j) = \sum_{n \in frame_j} \left| s(n) \right| \qquad (2.25)$$

และจะมีหลักการในการตัดหัวท้ายหน่วยที่ใช้ในการรู้จำคล้ายกับวิธีแรกคือ เมื่อสัญญาณมีค่า $E(j)$ มากกว่าค่าที่กำหนดติดกันมากกว่าจำนวนส่วนย่อยที่กำหนด (Frame Duration Threshold) ถือว่าเป็นช่วงเริ่มต้นของคำ และใช้หลักการเดียวกันในการหาจุดสิ้นสุดของคำ ข้อดีของการใช้ค่าแอมปลิจูดในการตัดหัวท้ายหน่วยที่ใช้ในการรู้จำคือ คำนวณง่าย ใช้เวลาน้อย แต่มีข้อเสียคือ มีโอกาสตัดผิดพลาดได้ ถ้ามีสัญญาณรบกวนที่มีค่าแอมปลิจูดสูง

2) ใช้ค่าพลังงาน (Energy)

มีหลักการเช่นเดียวกับการใช้แอมปลิจูด แต่เปลี่ยน $E(j)$ สำหรับส่วนย่อยที่ j ใดๆ เป็นค่าพลังงาน ซึ่งทำให้สัญญาณช่วงที่มีเสียงดังมีค่า $E(j)$ สูงขึ้นกว่าการใช้ค่าสัมบูรณ์ของแอมปลิจูด ส่งผลให้การตรวจสอบระดับของ $E(j)$ เพื่อตัดหัวท้าย หรือระหว่างพยางค์มีความถูกต้องมากขึ้น มีงานวิจัยที่เสนอสมการในการคำนวณค่าพลังงานของแต่ละส่วนย่อยหลายวิธี เช่น

- พลังงานแบบรากที่สองของกำลังสองเฉลี่ย (Root Mean Square Energy)

$$E(j) = \sqrt{\frac{\sum_{n=1}^{N} [s(n)]^2}{N}} \quad ; n = 1, 2, ..., N \quad , s(n) \in frame_j \qquad (2.26)$$

- พลังงานแบบผลรวมของกำลังสอง (Sum of Square Energy)

$$E(j) = \sum_{n=1}^{N} [s(n)]^2 \qquad (2.27)$$

หอสมุดกลาง สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

- พลังงานแบบ Teager

$$E(n) = [s(n)]^2 - (n+1)[s(n-1)]$$
$$E(j) = \sum_{n=1}^{N} E(n) \qquad (2.28)$$

3) ค่าอื่นๆ ที่ช่วยในการปรับแต่งจุดตัด

นอกเหนือจากการใช้ค่าแอมปลิจูด หรือพลังงานในการหาขอบของคำ หรือพยางค์ ยังมีการใช้ค่าอื่นๆ มาช่วยเพิ่มความถูกต้องของการหาอีก ได้แก่

- ความถี่มูลฐาน (Fundamental Frequency) เนื่องจากสัญญาณเสียงในช่วงที่เป็นเสียงสระ (Vowel) จะมีลักษณะเป็นรายคาบ (Periodic) (ณัฐกร ทับทอง 2538) ซึ่งจะมีค่าความถี่มูลฐาน และเสียงพยางค์ในภาษาไทยจะประกอบด้วยส่วนที่เป็นสระเสมอ ดังนั้นช่วงของความถี่มูลฐานช่วงหนึ่งจะตรงกับพยางค์หนึ่งพยางค์

- อัตราการตัดศูนย์ (Zero-crossing Rate) คือจำนวนครั้งของสัญญาณเสียงที่ตัดแกนศูนย์ในช่วงเวลาใดๆ ปกติสัญญาณเสียงในช่วงที่เงียบจะมีอัตราการตัดศูนย์สูงกว่าช่วงที่มีเสียงดัง (ณัฐกร ทับทอง 2538)

- ระยะเวลาแปล่งเสียง (Time Duration) เราสามารถเก็บข้อมูลเพื่อตรวจสอบว่าพยางค์ในภาษาไทยควรมีระยะเวลาในการเปล่งเสียงนานเท่าใด เป็นส่วนช่วยเสริมความถูกต้องของการตัดแบ่งพยางค์

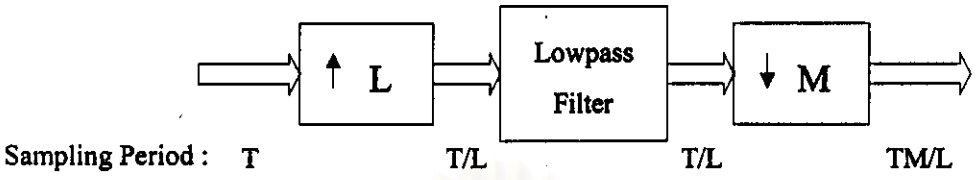
2.5.2 การนอร์มอลไลซ์ทางเวลา (Time Normalization)

เนื่องจากจำนวนโหนดในระดับชั้นข้อมูลเข้าของนิวรอลเน็ตเวอร์กแบบ MLP มีค่าคงที่ค่าหนึ่งเสมอ ดังนั้นเวกเตอร์ข้อมูลเข้าที่สกัดมาจากสัญญาณเสียงของคำที่จะรู้จำควรจะต้องมีมิติเท่ากันทุกคำ วิธีที่จะทำให้สัญญาณเสียงมีความยาวเท่ากันคือ การนอร์มอลไลซ์ทางเวลา (Time Normalization) ซึ่งมีวิธีการหลักๆ ดังนี้

1) การประมาณค่าโดยใช้การเปลี่ยนอัตราการชักตัวอย่าง (Sampling Rate)

การนอร์มอลไลซ์ทางเวลาด้วยวิธีนี้ แสดงได้ดังรูปที่ 2.10 โดยจะนำสัญญาณเสียงมาเพิ่มอัตราการชักตัวอย่างขึ้น L เท่า โดยการเพิ่มค่าศูนย์ (Zero-Packing) ระหว่างแต่ละจุดสัญญาณเดิม จากนั้นนำสัญญาณที่ได้ไปผ่านวงจรกรองแบบผ่านความถี่ต่ำ (Lowpass Filter) แล้วลดอัตราการชักตัวอย่างลง M เท่า โดย M เป็นจำนวนจุดสัญญาณเดิมก่อนการทำนอร์มอลไลซ์ทางเวลา และ

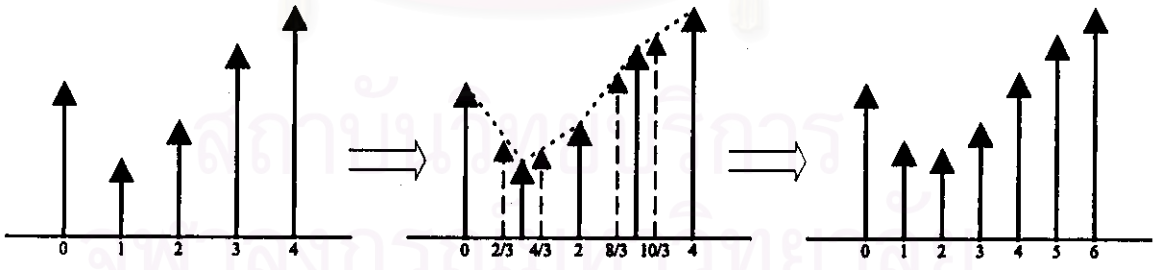
L เป็นจำนวนจุดสัญญาณที่ต้องการ ข้อดีของวิธีนี้คือ ไม่ทำให้เกิดการเคลือบแฝง (Aliasing) ได้ ข้อเสียคือ ใช้หน่วยความจำ และเวลาในการคำนวณมาก



รูปที่ 2.10 การนอร์มอลไลซ์ทางเวลาโดยใช้การเปลี่ยนอัตราการชักตัวอย่าง

2) การประมาณค่าในช่วงเชิงเส้น (Linear Interpolation)

จะทำการประมาณค่าแอมปลิจูดของสัญญาณที่จุดที่ไม่ทราบค่า จากความสัมพันธ์แบบเชิงเส้นของจุดสัญญาณเดิมก่อนการนอร์มอลไลซ์ทางเวลา ที่อยู่รอบจุดที่ไม่ทราบค่านั้น ตัวอย่างการนอร์มอลไลซ์ทางเวลาด้วยวิธีนี้ แสดงได้ดังรูปที่ 2.11 รูปที่ 2.11(ก) เป็นสัญญาณขนาด 5 จุดข้อมูล แต่ถ้าเราต้องการนอร์มอลไลซ์สัญญาณดังกล่าวให้เป็นสัญญาณขนาด 7 จุดข้อมูล เราจะทำการคำนวณค่าแอมปลิจูดของจุดข้อมูลใหม่ โดยเลื่อนช่วงเวลาไปทีละ 2 ใน 3 ของช่วงเวลาเดิม และใช้วิธีประมาณค่าในช่วงเชิงเส้นระหว่างค่าแอมปลิจูดของจุดข้อมูลเดิมที่อยู่ข้างๆ จุดนี้ ดังแสดงในรูปที่ 2.11(ข) เส้นประคือ แอมปลิจูดของจุดข้อมูลใหม่ที่ได้จากการประมาณค่า ผลลัพธ์สุดท้ายจะได้สัญญาณเสียงใหม่ที่ผ่านการนอร์มอลไลซ์ทางเวลา มีจำนวนจุดข้อมูลตามที่ต้องการ ดังแสดงในรูปที่ 2.11(ค)



(ก) สัญญาณเสียงก่อนการนอร์มอลไลซ์ทางเวลา

(ข) การประมาณค่าในช่วงเชิงเส้น

(ค) สัญญาณเสียงที่ผ่านการนอร์มอลไลซ์ทางเวลา

รูปที่ 2.11 ตัวอย่างการนอร์มอลไลซ์ทางเวลาโดยการประมาณค่าในช่วงเชิงเส้น

วิธีการนี้มีข้อดีคือ คำนวณง่าย ใช้เวลาในการคำนวณน้อย แต่มีข้อเสียคือ ทำให้เกิดการเคลือบแฝง (Aliasing) ในสัญญาณที่ประมาณค่า กล่าวคือ สัญญาณที่ผ่านการนอร์มอลไลซ์ด้วยวิธีนี้แล้ว จะให้เสียงที่มีเสียงแทรกตลอดเวลา อย่างไรก็ตาม สัญญาณที่ได้นี้ยังคงมีลักษณะเฉพาะตัวที่นำไปใช้ในการรู้จำได้เหมือนปกติ

2.5.3 การสกัดลักษณะเด่นของสัญญาณเสียง (Feature Extraction)

การใช้สัญญาณเสียงมาเปรียบเทียบกับคำอ้างอิง (Word Reference) แบบจุดต่อจุดโดยตรง ทำให้ต้องเสียเวลาในการคำนวณ และสิ้นเปลืองเนื้อที่หน่วยความจำมาก การเปรียบเทียบเฉพาะจุดสำคัญ หรือจุดเด่นของสัญญาณเสียงจะช่วยลดการคำนวณลงได้มาก การสกัดค่าลักษณะเด่นที่เหมาะสมของสัญญาณเสียง จึงนับเป็นขั้นตอนที่สำคัญมาก เพราะถ้าสามารถสกัดค่าลักษณะเด่นที่บ่งบอกความแตกต่างระหว่างสัญญาณเสียงแต่ละเสียงได้ดี อัตราการรู้จำก็จะสูงขึ้น ในขณะเดียวกัน ถ้ายิ่งใช้จำนวนค่าลักษณะเด่นน้อย การคำนวณเพื่อฝึกฝน และรู้จำจะยิ่งลดลง

โดยทั่วไป ลักษณะเด่นที่สกัดจากสัญญาณเสียง จะเป็นลักษณะเด่นเชิงสเปกตรัม (Spectral Feature) เนื่องจากลักษณะเด่นส่วนใหญ่ สำหรับการรู้จำเสียงโดยหูของมนุษย์ รวมอยู่ในข้อมูลเชิงสเปกตรัม อีกทั้ง สัญญาณเสียงมักถูกมองเป็นการรวมกันของสัญญาณ Sinusoidal หลายๆ ความถี่ ค่าเชิงสเปกตรัมที่จะพิจารณาคือ ความหนาแน่นของสเปกตรัมพลังงาน (Power Spectral Density) ของสัญญาณเสียงช่วงเวลาสั้นๆ (Short-time Interval) ซึ่งเป็นผลคูณระหว่าง 2 ส่วนคือ

1) เอนเวโลปเชิงสเปกตรัม (Spectral Envelope) เป็นค่าที่มีการเปลี่ยนแปลงน้อยเมื่อความถี่เปลี่ยนไป เป็นค่าที่บ่งบอกถึงลักษณะหลายๆ อย่างเกี่ยวกับสัญญาณเสียงช่วงนั้น เช่น ลักษณะการกำทอน (Resonance Characteristic) และลักษณะการแผ่เสียง (Radiation Characteristic) ของอวัยวะที่ออกเสียง

2) โครงสร้างย่อยเชิงสเปกตรัม (Spectral Fine Structure) เป็นค่าที่มีการเปลี่ยนแปลงอย่างรวดเร็วเมื่อความถี่เปลี่ยนไป มีค่าสอดคล้องกับความเป็นรายคาบของเสียง (Periodicity) โดยจะมีโครงสร้างเป็นรายคาบ สำหรับเสียงก้อง (Voiced Sound) และมีโครงสร้างไม่เป็นรายคาบสำหรับเสียงไม่ก้อง (Unvoiced Sound)

สำหรับการสกัดลักษณะเด่นโดยรวมของสัญญาณเสียง จะสกัดจากเอนเวโลปเชิงสเปกตรัม โดยแบ่งวิธีการสกัดได้เป็น 2 แนวทางคือ การวิเคราะห์โดยใช้พารามิเตอร์ (Parameter Analysis) และการวิเคราะห์โดยไม่ใช้พารามิเตอร์ (Non-parameter Analysis)

การวิเคราะห์โดยใช้พารามิเตอร์ เป็นการวิเคราะห์โดยเลือกแบบจำลองที่เหมาะสมกับสัญญาณ แล้วทำการปรับแต่งค่าพารามิเตอร์ลักษณะเด่นที่ใช้ในแบบจำลองนั้น มีวิธีหลักๆ ดังนี้

1) การวิเคราะห์โดยการสังเคราะห์ (Analysis-by-Synthesis) วิธีนี้สามารถสร้างแบบจำลองได้แม่นยำ พารามิเตอร์ที่ใช้ได้แก่ ค่าความถี่ฟอร์แมนท์ (Formant) ความกว้างแถบความถี่ (Bandwidth) และเอนเวโลปเชิงสเปกตรัม แต่มีข้อเสียคือ ต้องใช้เวลาในการคำนวณมาก เพราะค่าพารามิเตอร์แต่ละตัวอาจมีผลต่อกัน ทำให้ต้องมีการคำนวณแบบวนซ้ำหลายครั้ง

2) การประมาณพันธะเชิงเส้น (Linear Predictive Coding) (LPC) เป็นการสร้างแบบจำลองของสเปกตรัมอย่างง่ายโดยใช้โพล (All-pole Spectrum Modeling) พารามิเตอร์สามารถประมาณค่าได้จากค่าความแปรปรวนร่วม (Covariance) หรือค่าอัตสหสัมพันธ์ (Autocorrelation) ข้อดีของวิธีนี้คือ สามารถแทนสัญญาณเสียงได้อย่างมีประสิทธิภาพโดยใช้พารามิเตอร์จำนวนน้อย และใช้การคำนวณที่ค่อนข้างง่าย

สำหรับการวิเคราะห์โดยไม่ใช้พารามิเตอร์ เป็นวิธีที่สามารถประยุกต์ใช้กับสัญญาณหลายชนิดได้ เพราะไม่ได้สร้างแบบจำลองสัญญาณ มีวิธีการหลักๆ ดังนี้

1) การวิเคราะห์โดยใช้เซปสตรัม (Cepstrum) มีข้อดีคือ สามารถแยกเอนเวโลปเชิงสเปกตรัม และโครงสร้างย่อยเชิงสเปกตรัมออกจากกันได้ในโดเมนแบบคิวเฟรินซี (Quefrency Domain) ซึ่งเป็นโดเมนทางเวลา แต่มีข้อเสียคือ ต้องคำนวณผลการแปลงฟูริเยร์แบบเร็ว (Fast Fourier Tranform) สองครั้ง ซึ่งใช้เวลาคำนวณมาก อย่างไรก็ตาม พารามิเตอร์ของเซปสตรัมยังสามารถคำนวณได้จากพารามิเตอร์ของการประมาณพันธะเชิงเส้นอีกทีหนึ่ง

2) ชุดวงจรกรองผ่านแถบความถี่ (Bandpass Filter Bank) วิธีนี้นำสัญญาณเสียงมาผ่านวงจรกรองแบบผ่านแถบความถี่หลายวงจร โดยมีช่วงความถี่ที่ผ่านได้แตกต่างกัน วงจรกรองแต่ละวงจรจะให้สัญญาณขาออก ที่สัมพันธ์กับค่าพลังงานของสัญญาณในช่วงความถี่ผ่านของวงจรกรองนั้นๆ ข้อดีของวิธีนี้คือ สร้างเป็นอุปกรณ์จริงได้ง่าย และเหมาะสมสำหรับการประมวลผลเวลาจริง (Real-time Processing)

3) การวิเคราะห์การตัดศูนย์ (Zero-crossing Analysis) โดยการนับจำนวนครั้งที่สัญญาณเสียงตัดแกนศูนย์ ซึ่งเป็นการประมาณค่าความถี่ฟอร์แมนท์ การวิเคราะห์นี้มักใช้ร่วมกับวิธีอื่นๆ ช่วยเพิ่มประสิทธิภาพในการสกัดค่าลักษณะเด่น

4) การวิเคราะห์โดยใช้ค่าอัตสหสัมพันธ์ในช่วงเวลาสั้นๆ (Short-time Autocorrelation) เป็นการทำคอนโวลูชัน (Convolution) ระหว่างเอนเวโลปเชิงสเปกตรัม กับโครงสร้างย่อยเชิงสเปกตรัม ข้อดีคือ คำนวณได้ง่าย และสามารถสร้างเป็นอุปกรณ์จริงได้ง่าย

2.5.4 การสกัดลักษณะเด่นด้วยการประมาณพันธะเชิงเส้น (LPC)

เนื่องจากงานวิจัยนี้ เลือกใช้วิธีการสกัดค่าลักษณะเด่นคือ การประมาณพันธะเชิงเส้น ซึ่งเป็นวิธีที่ใช้กันอย่างแพร่หลายในการหาค่าลักษณะเด่นของสัญญาณเสียง ดังนั้นจะขอกล่าวเฉพาะรายละเอียดของวิธีการนี้เท่านั้น ดังนี้

หลักการเบื้องต้นคือ จะสมมติให้แบบจำลองการสร้างสัญญาณเสียง ประกอบด้วยแหล่งกำเนิดสัญญาณกระตุ้น $U(z)$ ป้อนเข้าสู่วงจรกรองจัดรูปสัญญาณ $H(z)$ ทำให้ได้สัญญาณเสียง $S(z)=U(z)H(z)$ วงจรกรอง $H(z)$ สามารถจำลองได้โดยใช้ค่าสัมประสิทธิ์ ดังนี้

$$H(z)=\frac{S(z)}{U(z)}=G\frac{1+\sum_{l=1}^{q}b_l z^{-l}}{1-\sum_{k=1}^{p}a_k z^{-k}} \qquad (2.29)$$

การประมาณพันธะเชิงเส้น โดยส่วนใหญ่นิยมใช้แบบจำลองที่มีโพลเพียงอย่างเดียว (All-pole Model) ซึ่งมีแบบจำลองของวงจรกรองดังนี้

$$H(z)=\frac{S(z)}{U(z)}=\frac{G}{1-\sum_{k=1}^{p}a_k z^{-k}} \qquad (2.30)$$

ถ้าป้อนสัญญาณ $s(n)$ เข้าสู่วงจรกรอง Predictor ซึ่งเป็นส่วนกลับของ $H(z)$ ดังนี้

$$A(z)=1-\sum_{k=1}^{p}a_k z^{-k} \qquad (2.31)$$

สัญญาณขาออก จะเรียกว่าสัญญาณค่าความผิดพลาด $e(n)$ แสดงได้โดย

$$e(n)=s(n)-\sum_{k=1}^{p}a_k s(n-k) \qquad (2.32)$$

การประมาณพันธะเชิงเส้นโดยวิธีอัตสหสัมพันธ์ จะทำการคูณสัญญาณด้วยฟังก์ชันหน้าต่าง (Window Function) เพื่อจำกัดให้สัญญาณ $x(n)=w(n)s(n)$ มีค่าอยู่ในช่วงเวลา

จำกัด ซึ่งมีจำนวนข้อมูล N จุด ดังนั้น $x(n)=0$ เมื่อ $n<0$ และ $n\geq N$ แล้วคำนวณค่าสัมประสิทธิ์การประมาณพันธะเชิงเส้น (Linear Prediction Coefficient) เพื่อใช้เป็นตัวแทนของสัญญาณในแต่ละช่วงเวลานี้ กำหนดให้ E เป็นค่าพลังงานของความผิดพลาด

$$E=\sum_{-\infty}^{\infty}e^2(n)=\sum_{-\infty}^{\infty}[x(n)-\sum_{k=1}^{p}a_k x(n-k)]^2 \qquad (2.33)$$

ค่าสัมประสิทธิ์ a_k ที่ทำให้ E มีค่าน้อยที่สุด สามารถหาได้โดยการแก้สมการ $\frac{\partial E}{\partial a_k}=0$ เมื่อ $k=1,2,3,...,p$ ซึ่งจะได้สมการเชิงเส้น p สมการ ดังนี้

$$\sum_{n=-\infty}^{\infty}x(n-i)x(n)=\sum_{k=1}^{p}a_k\sum_{n=-\infty}^{\infty}x(n-i)x(n-k) \quad \text{เมื่อ } i=1,2,3,...,p \qquad (2.34)$$

เนื่องจากพจน์ทางด้านซ้ายมือของสมการคือ ค่าอัตสหสัมพันธ์ $R(i)$ ของ $x(n)$ โดยที่ $x(n)$ มีค่าในช่วงเวลาที่จำกัด ดังนั้นจะได้

$$\sum_{k=1}^{p}a_k R(i-k)=R(i) \quad \text{โดยที่} \quad R(i)=\sum_{n=i}^{N-1}x(n)x(n-i) \qquad (2.35)$$

เมื่อแก้สมการที่ 2.35 ซึ่งประกอบด้วยสมการเชิงเส้น p สมการ จะได้ค่าสัมประสิทธิ์ a_k ที่ใช้ในการประเมินค่าสัญญาณจำนวน p ค่า เมื่อมองสมการที่ 2.35 ในรูปของเมตริกซ์ จะเขียนได้ดังนี้

$$\begin{bmatrix} R_0 & R_1 & \cdots & R_{1-p} \\ R_1 & R_0 & \cdots & R_{2-p} \\ \vdots & \vdots & \ddots & \vdots \\ R_{p-1} & R_{p-2} & \cdots & R_0 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix}=\begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_p \end{bmatrix} \qquad (2.36)$$

เมตริกซ์ของค่าอัตสหสัมพันธ์ อยู่ในรูปของเมตริกซ์ Toeplitz ซึ่งมีลักษณะสมมาตร และทุกๆ สมาชิกในแนวทแยงมุมมีค่าเท่ากัน ทำให้การแก้สมการเพื่อหาค่าสัมประสิทธิ์

a_k โดยวิธีนี้ง่ายกว่าการใช้วิธีความแปรปรวนร่วม (Covariance) วิธีหนึ่งที่เหมาะสมในการแก้สมการที่อยู่ในรูปเมตริกซ์ Toeplitz คือ วิธีของ Levinson-Durbin (O'Shaughnessy, 1988) โดยการคำนวณลำดับของสมการสำหรับ $m = 1, 2, 3, ..., p$ ดังนี้

$$
\begin{aligned}
k_m &= \frac{R(m) - \sum_{i=1}^{m-1} a_{m-1} R(m-i)}{E_{m-1}} \\
a_m(m) &= k_m \\
a_m(i) &= a_{m-1}(i) - k_m a_{m-1}(m-i) \qquad ; 1 \le i < m \\
E_m &= (1 - k_m{}^2) E_{m-1}
\end{aligned}
\tag{2.37}
$$

โดยที่ $E_0 = R(0)$ และ $a_0 = 0$ ในแต่ละรอบของ m ค่าสัมประสิทธิ์ $a_m(i)$ เมื่อ $i = 1, 2, 3, ..., m$ จะเป็นค่าการประมาณพันธะเชิงเส้นอันดับที่ m ที่เหมาะสมที่สุด

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย