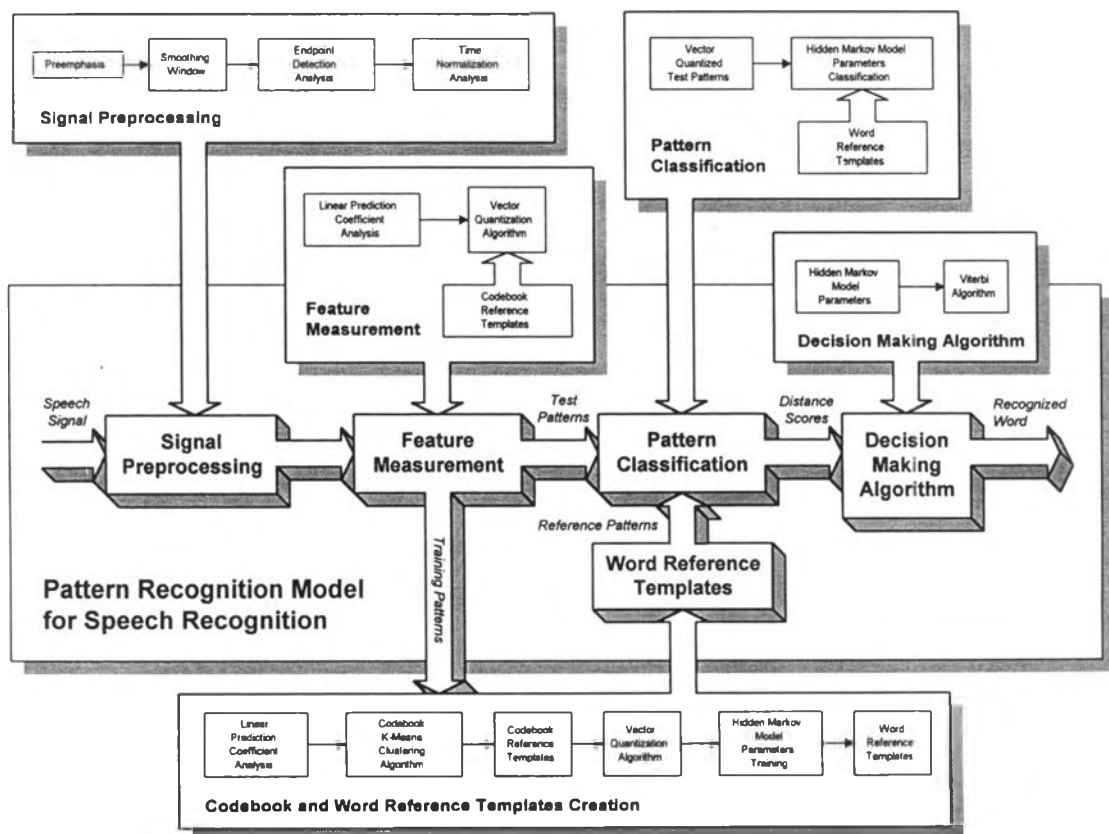


ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

รายละเอียดในบทที่ 2 นี้ จะกล่าวถึงทฤษฎีต่างๆ ที่เกี่ยวข้องและอ้างอิงถึงในงานวิจัยนี้ โดยแบ่งออกเป็น 3 ขั้นตอนดังแสดงในรูปที่ 1.1 และ 2.1 (Rabiner and Levinson, 1981; Roe and Wilpon, 1993) โดยขั้นตอนแรกเริ่มจากทฤษฎีที่เกี่ยวข้องกับการประมวลผลสัญญาณเบื้องต้น (Signal Preprocessing) ขั้นตอนที่สองเป็นขั้นตอนการวิเคราะห์และวัดค่าลักษณะสำคัญ (Feature Measurement) เกี่ยวข้องกับทฤษฎีของสัมประสิทธิ์ของการประมาณฟังก์ชันเชิงเส้นและขั้นตอนวิธีการควอนไทซ์แบบเวกเตอร์ ขั้นตอนที่สามเป็นการจำแนกรูปแบบ (Pattern Classification) เกี่ยวข้องกับทฤษฎีของการเปรียบเทียบรูปแบบที่นำมาทดสอบกับต้นแบบที่มีอยู่ โดยใช้หลักการของแบบจำลองฮิดเดน มาร์คอฟร่วมกับการควอนไทซ์แบบเวกเตอร์ และขั้นตอนสุดท้ายเป็นขั้นตอนวิธีการตัดสินใจ (Decision Making Algorithm) ในการเลือกต้นแบบที่ตรงกับรูปแบบที่นำมาทดสอบมากที่สุดโดยอาศัยขั้นตอนวิธีการของ Viterbi (Viterbi Algorithm)



รูปที่ 2.1 รายละเอียดของแบบจำลองรูปแบบการรู้จำคำพูดภาษาไทยเป็นคำหลายพยางค์

ก. การประมวลผลสัญญาณเบื้องต้น (Signal Preprocessing)

การประมวลผลสัญญาณเบื้องต้นเป็นขั้นตอนกรรมวิธีในการจัดเตรียมข้อมูล จากข้อมูลดิบของเสียงพูดที่ได้จากการบันทึกเสียงนำมาผ่านกรรมวิธีประมวลผลสัญญาณเชิงเลข เพื่อใช้ในการประมวลผลในขั้นตอนต่อไป เนื่องจากสัญญาณเสียงพูดโดยรวมจะแปรเปลี่ยนตามเวลาและไม่เสถียร (Nonstationary) ดังนั้นในการประยุกต์ใช้งานกรรมวิธีประมวลผลสัญญาณเชิงเลขกับสัญญาณเสียงพูด จึงต้องแบ่งสัญญาณเสียงพูดออกเป็นส่วนย่อย (Rabiner and Levinson, 1981; Furui, 1985) เรียกว่า "กรอบเสียงพูด" (Speech Frame) โดยแต่ละกรอบเสียงพูดจะมีความยาวประมาณ 10 - 40 มิลลิวินาที ซึ่งถือได้ว่าสัญญาณเสียงพูดในแต่ละกรอบเสียงพูดมีความเสถียรและไม่แปรเปลี่ยนตามเวลา (Stationary) จากนั้นจึงสามารถทำการประมวลผลสัญญาณเชิงเลขกับสัญญาณเสียงพูดในแต่ละกรอบเสียงพูดได้

กรรมวิธีในการประมวลผลสัญญาณเบื้องต้นแบ่งได้เป็น 4 กรรมวิธีย่อยดังแสดงในรูปที่ 2.1 ได้แก่ กรรมวิธีเน้นล่วงหน้า (Preemphasis) กรรมวิธีวางกรอบขนาดสัญญาณ (Smoothing Window) กรรมวิธีหาจุดสิ้นสุดเสียงพูด (Endpoint Detection) และกรรมวิธีปรับบรรทัดฐานเชิงเวลา (Time Normalization) ตามลำดับ

1. กรรมวิธีเน้นล่วงหน้า (Preemphasis)

ขั้นตอนกรรมวิธีเน้นล่วงหน้าเป็นการบีบอัดช่วงพิสัยพลวัต (Dynamic Range) ของสัญญาณเสียงพูด โดยการทำให้ความลาดเอียงในเชิงความถี่แบนราบลง ซึ่งจะส่งผลให้ค่าอัตราส่วนสัญญาณต่อสัญญาณรบกวนมีค่าสูงขึ้น ในทางปฏิบัติแล้วจะนำสัญญาณผ่านตัวกรองเชิงเลขลำดับหนึ่ง (First-Order Digital Filter) ที่มีฟังก์ชันถ่ายโอนดังแสดงในสมการที่ (2.1) และ (2.2) (Furui, 1985) เมื่อ a เป็นสัมประสิทธิ์ของตัวกรอง $\tilde{s}(n)$ เป็นค่าของสัญญาณเสียงพูดขาออกที่ผ่านกรรมวิธีเน้นล่วงหน้าที่ n $s(n)$ เป็นค่าของสัญญาณเสียงพูดขาเข้าที่ n และ $s(n-1)$ เป็นค่าของสัญญาณเสียงพูดขาเข้าค่าก่อนหน้าที่ $n-1$ ดังนี้

$$H(z) = 1 - az^{-1} \dots\dots\dots (2.1)$$

$$\tilde{s}(n) = s(n) - as(n-1) \dots\dots\dots (2.2)$$

โดยกำหนดให้ค่าสัมประสิทธิ์ของตัวกรอง a มีค่าเข้าใกล้ 1 เมื่อใช้ร่วมกับการวิเคราะห์หาค่าสัมประสิทธิ์การประมาณพัลเซเชิงเส้นจะกำหนดให้ค่า $a = 0.95$ เนื่องจากเป็นค่าที่ให้ผลดีที่สุดสำหรับการวิเคราะห์ (Rabiner, Levinson, Rosenberg, and Wilpon, 1979)

2. กรรมวิธีวางกรอบขนาดสัญญาณ (Smoothing Window)

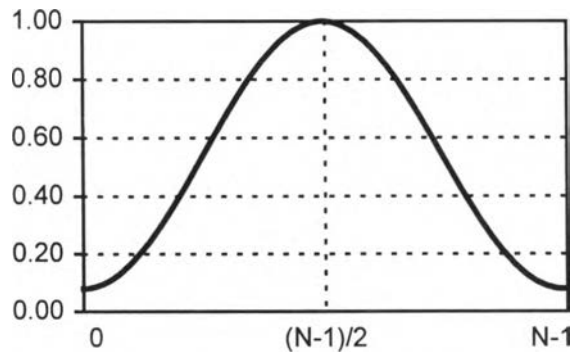
ขั้นตอนกรรมวิธีวางกรอบขนาดสัญญาณจัดเป็นขั้นตอนในการเตรียมข้อมูลในแต่ละกรอบข้อมูลเสียงพูดเพื่อการวิเคราะห์อัตโนมัติ โดยการคูณแต่ละค่าของสัญญาณในกรอบข้อมูลเสียงพูดด้วยค่า

ฟังก์ชันกรอบ (Window Function) ซึ่งมีหลายประเภทได้แก่ Rectangular Window, Hamming Window, Hanning Window, Blackman Window, Kaiser Window เป็นต้น (Oppenheim and Schaffer, 1989) ผลของการวางกรอบขนาดสัญญาณมี 2 ประการ ประการแรก เป็นการลดทอนแอมพลิจูดอย่างช้าๆ ที่บริเวณปลายแต่ละข้างของกรอบข้อมูลเสียงพูดเพื่อป้องกันการเปลี่ยนแปลงอย่างกะทันหันที่จุดปลาย ประการที่สอง เป็นการสร้างค่าการประสานสำหรับการผลแปลงฟูรีเยร์ของฟังก์ชันกรอบและแถบสเปกตรัมของเสียงพูด สำหรับการวิเคราะห์เสียงพูดจะใช้ฟังก์ชันกรอบชนิด Hamming Window ดังแสดงในรูปที่ 2.2 ซึ่งใช้ในการวิเคราะห์เสียงพูดโดยเฉพาะดังแสดงในสมการที่ (2.3) และ (2.4) เมื่อ L เป็นจำนวนกรอบข้อมูลเสียงพูดทั้งหมด N เป็นจำนวนข้อมูลในแต่ละกรอบข้อมูลเสียงพูด l เป็นกรอบที่ l ของกรอบ L ทั้งหมด และ n เป็นข้อมูลที่ n ของข้อมูลทั้งหมด N ค่าซึ่งอยู่ภายในกรอบที่ l (Furui, 1985; Oppenheim and Schaffer, 1989)

$$\tilde{x}_l(n) = x_l(n) \cdot w(n) \dots\dots\dots (2.3)$$

$$w(n) = 0.54 - 0.46 \cos \left[\frac{2\pi n}{N-1} \right] \dots\dots\dots (2.4)$$

เมื่อ $l = 0, 1, \dots, L-1$ $n = 0, 1, \dots, N-1$

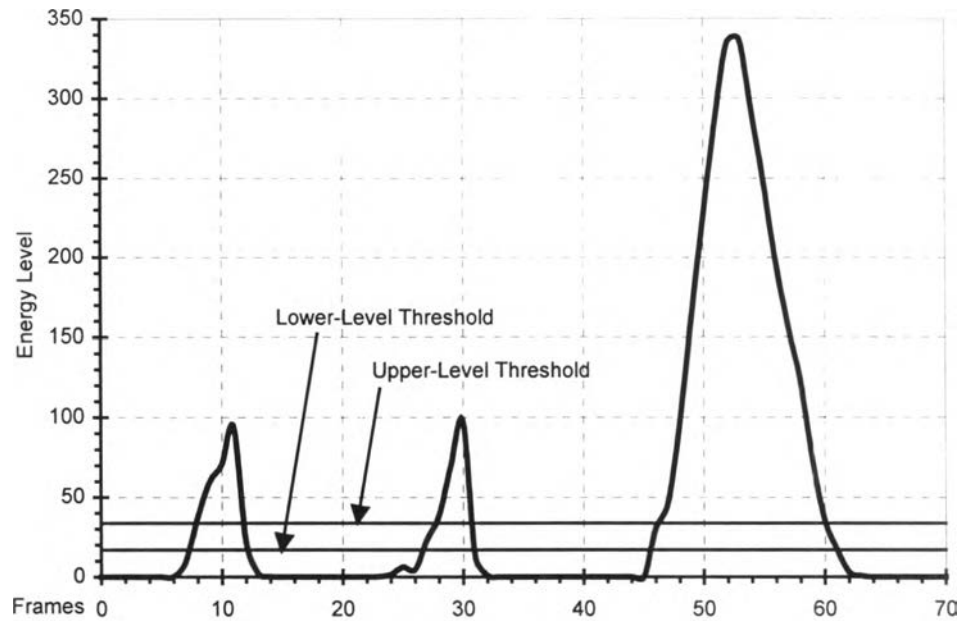


รูปที่ 2.2 ฟังก์ชันกรอบชนิด Hamming Window

3. กรรมวิธีหาจุดสิ้นสุดเสียงพูด (Endpoint Detection)

ขั้นตอนกรรมวิธีหาจุดสิ้นสุดเสียงพูดนี้จัดได้ว่าเป็นขั้นตอนที่สำคัญที่สุดขั้นตอนหนึ่งในกระบวนการรู้จำคำพูด เนื่องจากการกำหนดจุดเริ่มต้นและจุดสิ้นสุดของคำทั้งหมดหรือของแต่ละพยางค์เพื่อการแยกแยะเสียงพูด เพื่อนำแต่เฉพาะส่วนที่เป็นเสียงพูดมาทำการวิเคราะห์ในขั้นตอนการรู้จำเสียงพูดของคำศัพท์แต่ละคำ สำหรับงานวิจัยนี้จะอาศัยแผนภูมิเส้นระดับพลังงาน (Energy Level Contour) ของเสียงพูดโดยการวิเคราะห์ค่าระดับพลังงานของแต่ละกรอบข้อมูลเสียงพูด ดังแสดงในสมการที่ (2.10) เมื่อ $E(m)$ เป็นค่าระดับพลังงานของกรอบข้อมูลเสียงพูดที่ m และ $s(n)$ แทนค่าของสัญญาณค่าที่ n ในกรอบข้อมูลเสียงพูด (Rabiner and Levinson, 1981)

$$E(m) = \sum_{n=0}^{N-1} |s(n)| \dots\dots\dots (2.10)$$



รูปที่ 2.3 แผนภูมิเส้นระดับพลังงาน

ขั้นตอนกรรมวิธีการหาจุดสิ้นสุดเสียงพูดอาศัยการวิเคราะห์ค่าระดับพลังงานของเสียงพูดแต่ละค่าดังแสดงในสมการที่ (2.10) และในรูปที่ 2.3 โดยการกำหนดจุดเริ่มเปลี่ยนของระดับพลังงาน (Energy Level Threshold) จำนวน 2 ระดับที่ระดับพลังงานขั้นต่ำ (Lower-Level Threshold) และระดับพลังงานขั้นสูง (Upper-Level Threshold) โดยกำหนดเป็นร้อยละของระดับพลังงานสูงสุดตามลำดับ ร่วมกับช่วงเวลาของระดับพลังงาน (Energy Level Duration) ของเส้นระดับพลังงานที่สูงกว่าระดับพลังงานขั้นสูง

การหาจุดเริ่มต้นและจุดสิ้นสุดของเสียงพูด กระทำโดยค่าระดับพลังงานที่มีค่าสูงเกินกว่าระดับพลังงานขั้นสูงต่อเนื่องกันเกินกว่าช่วงเวลาของระดับพลังงานที่กำหนด จึงจะนับกรอบข้อมูลเสียงพูดซึ่งมีจุดเริ่มเปลี่ยนที่ระดับพลังงานขั้นต่ำเป็นจุดเริ่มต้นของเสียงพูด จุดสิ้นสุดเสียงพูดเป็นกรอบข้อมูลเสียงพูดที่มีค่าระดับพลังงานต่ำกว่าระดับพลังงานขั้นต่ำ และจะต้องมีจุดเริ่มต้นของเสียงพูดเกิดขึ้นมาก่อนจึงจะเริ่มตรวจจับจุดสิ้นสุดเสียงพูด ดังแสดงในรูปที่ 2.3 การกำหนดระดับพลังงานเป็นร้อยละของค่าระดับพลังงานสูงสุดจึงไม่จำเป็นต้องทำให้ระดับพลังงานเป็นบรรทัดฐานเดียวกันทั้งหมด (Normalize) ก่อนการวิเคราะห์

4. กรรมวิธีปรับบรรทัดฐานเชิงเวลา (Time Normalization)

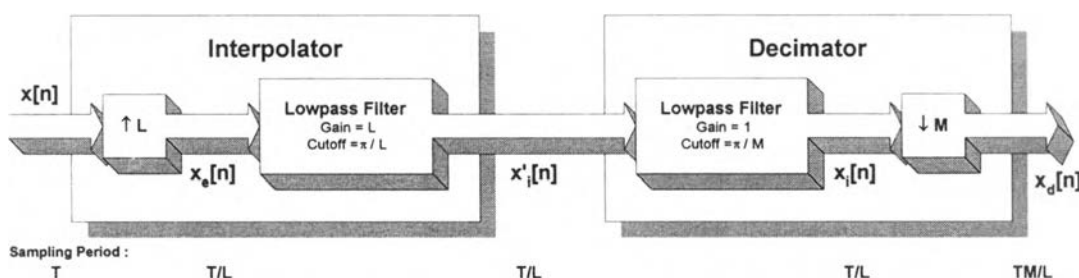
ขั้นตอนกรรมวิธีปรับบรรทัดฐานเชิงเวลาจัดเป็นขั้นตอนในการเพิ่มหรือลดขนาดความยาวของสัญญาณในเชิงเวลา เพื่อปรับแต่งขนาดความยาวของสัญญาณให้เหมาะสมเนื่องจากสัญญาณเสียงพูดที่ได้

จากเสียงพูดของแต่ละบุคคลมีความยาวไม่เท่ากัน จึงจำเป็นต้องมีการปรับขนาดความยาวของสัญญาณในเชิงเวลาให้มีขนาดเท่าที่กำหนดเพื่อใช้ในการหาค่าลักษณะสำคัญและเปรียบเทียบสัญญาณเสียงต่อไป

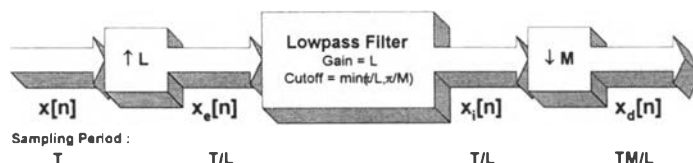
กรรมวิธีในการปรับบรรทัดฐานเชิงเวลาจะอาศัยขั้นตอนในการลดหรือเพิ่มอัตราการสุ่มตัวอย่าง (Sampling Rate) หรือความถี่ในการสุ่มตัวอย่าง (Sampling Frequency) หรือคาบเวลาในการสุ่มตัวอย่าง (Sampling Period) เพื่อปรับขนาดความยาวของสัญญาณในเชิงเวลาให้เป็นไปตามที่ต้องการ เนื่องจากการเปลี่ยนแปลงอัตราการสุ่มตัวอย่างของเสียงพูดนั้นเป็นการเปลี่ยนแปลงแบบไม่เป็นจำนวนเต็ม ดังนั้นการเปลี่ยนแปลงอัตราการสุ่มจึงต้องเพิ่มอัตราการสุ่มให้สูงขึ้นเป็นจำนวน L เท่า จากนั้นจึงจะลดอัตราการสุ่มลงเป็นจำนวน M เท่าตามต้องการ และเพื่อเป็นการป้องกันไม่ให้ข้อมูลเกิดการสูญหายเมื่อเปรียบเทียบกับการลดอัตราสุ่มลงเพียงอย่างเดียว โดยจำนวนเท่าของการเปลี่ยนแปลงจะเป็นอัตราส่วนระหว่างจำนวนเท่าของข้อมูลเสียงพูดที่ลดลงกับจำนวนเท่าของข้อมูลเสียงพูดที่เพิ่มขึ้นดังนี้ (Oppenheim and Schaffer, 1989)

$$T' = \frac{M}{L} T \dots\dots\dots (2.11)$$

เมื่อ T เป็นคาบเวลาในการสุ่มตัวอย่างของสัญญาณที่ต้องการเปลี่ยนแปลงความถี่ในการสุ่มตัวอย่าง T' เป็นคาบเวลาในการสุ่มตัวอย่างของสัญญาณที่ได้รับการเปลี่ยนแปลง M เป็นจำนวนเท่าของอัตราการสุ่มตัวอย่างที่เพิ่มขึ้น และ L เป็นจำนวนเท่าของอัตราการสุ่มตัวอย่างที่ลดลง โดยมีขั้นตอนกรรมวิธีดังแสดงในรูปที่ 2.4 และ 2.5 ดังนี้ (Oppenheim and Schaffer, 1989)



รูปที่ 2.4 ขั้นตอนกรรมวิธีในการเปลี่ยนแปลงอัตราการสุ่มตัวอย่างแบบไม่เป็นจำนวนเต็ม



รูปที่ 2.5 ขั้นตอนกรรมวิธีในการเปลี่ยนแปลงอัตราการสุ่มตัวอย่างเมื่อรวมตัวกรองแบบผ่านต่ำไว้ด้วยกัน

ขั้นตอนกรรมวิธีในการเปลี่ยนแปลงอัตราการสุ่มตัวอย่างแบบไม่เป็นจำนวนเต็มดังแสดงในรูปที่ 2.4 และ 2.5 นั้น เป็นการเพิ่มอัตราการสุ่มตัวอย่างให้สูงขึ้น L เท่าซึ่งเรียกว่า Upsampling หรือ

Interpolation จากนั้นจึงทำการลดอัตราการสุ่มตัวอย่างลง M เท่าซึ่งเรียกว่า Downsampling หรือ Decimation โดยมีรายละเอียดดังนี้ (Oppenheim and Schaffer, 1989)

4.1 การเพิ่มอัตราการสุ่มตัวอย่าง (Upsampling)

การเพิ่มอัตราการสุ่มตัวอย่างเริ่มจากการสุ่มตัวอย่างสัญญาณที่ต่อเนื่องทางเวลา (Continuous-time Signal) $x_c(nT)$ เมื่อ $x[n]$ เป็นสัญญาณที่สุ่มได้ดังนี้

$$x[n] = x_c(nT) \dots\dots\dots (2.12)$$

เมื่อทำการเพิ่มอัตราการสุ่มตัวอย่างขึ้น L เท่าด้วยการลดขนาดของคาบเวลาในการสุ่มตัวอย่าง T ลง L เท่า โดยที่ $T' = T/L$ จะได้ว่า

$$x_i[n] = x_c(nT') \dots\dots\dots (2.13)$$

ดังนั้น $x_i[n] = x[n/L] = x_c(nT/L)$, $n = 0, \pm L, \pm 2L, \dots$ (2.14)

เมื่อ $x_i[n]$ เป็นสัญญาณที่ได้รับการเพิ่มอัตราการสุ่มตัวอย่าง และเพื่อเป็นการลดผลของการเคลือบแฝง (Aliasing) ของสัญญาณอันเนื่องมาจากจำนวนตัวอย่างที่เพิ่มขึ้น จึงจำเป็นต้องทำการกรองสัญญาณด้วยตัวกรองแบบผ่านต่ำ (Lowpass Filter) ที่มีความถี่ตัด (Cutoff Frequency) ที่ตำแหน่ง π/L ภายหลังการเพิ่มอัตราการสุ่มตัวอย่างดังแสดงในรูปที่ 2.4 ดังนั้นระบบการเพิ่มอัตราการสุ่มตัวอย่างจึงประกอบไปด้วยตัวเพิ่มอัตราการสุ่มตัวอย่างและตัวกรองแบบผ่านต่ำ โดยเรียกระบบนี้ว่า Interpolator

4.2 การลดอัตราการสุ่มตัวอย่าง (Downsampling)

การลดอัตราการสุ่มตัวอย่างจะเป็นการเปลี่ยนแปลงลำดับการสุ่มตัวอย่างของทั้งสัญญาณที่สุ่มได้ $x[n]$ หรือสัญญาณที่ต่อเนื่องทางเวลา $x_c(nT)$ เพื่อให้เกิดลำดับใหม่ที่ต้องการขึ้นมาด้วยการเพิ่มขนาดของคาบเวลาในการสุ่มตัวอย่าง T เพิ่มขึ้น M เท่า โดยที่ $T' = MT$ และเมื่อ $x_d[n]$ เป็นสัญญาณที่สุ่มได้ดังนี้

$$x_d[n] = x[nM] = x_c(nMT) \dots\dots\dots (2.15)$$

เพื่อเป็นการลดผลของการเคลือบแฝง (Aliasing) ของสัญญาณอันเนื่องมาจากแถบความถี่ของสัญญาณที่เพิ่มขึ้น จึงจำเป็นต้องทำการลดขนาดแถบความถี่ของสัญญาณด้วยตัวกรองแบบผ่านต่ำ (Lowpass Filter) ที่มีความถี่ตัด (Cutoff Frequency) ที่ตำแหน่ง π/M ก่อนการลดอัตราการสุ่มตัวอย่างดังแสดงในรูปที่ 2.4 ดังนั้นระบบการเพิ่มอัตราการสุ่มตัวอย่างจึงประกอบไปด้วยตัวกรองแบบผ่านต่ำและตัวลดอัตราการสุ่มตัวอย่าง โดยเรียกระบบนี้ว่า Decimator

ดังนั้นในทางปฏิบัติเพื่อการเพิ่มหรือลดอัตราการสุ่มตัวอย่างแบบไม่เป็นจำนวนเต็ม จะใช้ระบบดังแสดงในรูปที่ 2.5 โดยรวมตัวกรองแบบผ่านต่ำของระบบ Interpolator และระบบ Decimator เข้าด้วยกันเป็นตัวเดียว โดยมีเงื่อนไขในการเลือกความถี่ตัดตามขนาดของ M และ L ที่มีค่าน้อยที่สุด

ข. การวิเคราะห์และวัดค่าลักษณะสำคัญ (Feature Measurement)

การวิเคราะห์และวัดค่าลักษณะสำคัญเป็นการวิเคราะห์สัญญาณเสียงพูด เพื่อเก็บรวบรวมลักษณะสำคัญของเสียงพูดแต่ละเสียง สำหรับการฝึกฝนระบบให้รับรู้ถึงความแตกต่างของเสียงพูดแต่ละเสียงและเพื่อใช้ในการเปรียบเทียบแยแ่กความแตกต่างของเสียงพูดแต่ละเสียงออกจากกัน ในการวัดค่าลักษณะสำคัญ จะใช้วิธีการหาสัมประสิทธิ์ของการประมาณพั้นระเชิงเส้น ส่วนการวิเคราะห์ลักษณะสำคัญจะอาศัยขั้นตอนการควอนไทซ์แบบเวกเตอร์เป็นหลัก

1. สัมประสิทธิ์ของการประมาณพั้นระเชิงเส้น (Linear Prediction Coefficient)

คำว่า "การประมาณพั้นระเชิงเส้น" หรือ Linear Prediction ถูกนำเสนอเป็นครั้งแรกโดย N. Wiener ในปี ค.ศ. 1966 โดยเทคนิคนี้ถูกนำมาใช้เป็นครั้งแรกกับการวิเคราะห์และการสังเคราะห์เสียงโดย Itakura กับ Saito และ Atal กับ Schroeder ในปี ค.ศ. 1968 (Furui, 1991) ความสำคัญของเทคนิคการประมาณพั้นระเชิงเส้นก็คือ การที่รูปคลื่นและลักษณะสมบัติทางความถี่ของเสียงพูดสามารถแสดงด้วยค่าพารามิเตอร์เพียงไม่กี่ค่าได้อย่างแม่นยำและมีประสิทธิภาพ นอกจากนี้ค่าพารามิเตอร์ดังกล่าวยังสามารถคำนวณได้ง่ายอีกด้วย

กำหนดให้สัญญาณเสียงที่ถูกสุ่มอย่างไม่ต่อเนื่องทุกเวลา ΔT วินาที แทนได้ด้วย $\{x(t)\}$ เมื่อ t เป็นจำนวนเต็ม และ $\Delta T \leq 1/2W$ วินาทีเมื่อความถี่ของสัญญาณเสียงอยู่ในช่วง $0 - W$ เฮิรตซ์ ดังนั้นความสัมพันธ์ระหว่างสัญญาณสุ่ม $x(t)$ กับสัญญาณสุ่มก่อนหน้านี p ค่าแสดงได้เป็น (Furui, 1991)

$$x_t + \alpha_1 x_{t-1} + \dots + \alpha_p x_{t-p} = \varepsilon_t \dots\dots\dots (2.16)$$

เมื่อ $\{\varepsilon_t\}$ เป็นตัวแปรทางสถิติชนิดอสหสัมพันธ์ ที่มีค่าเฉลี่ยเป็นศูนย์และมีค่าความแปรปรวนเป็น σ^2 ซึ่งจะสามารถประมาณสัญญาณสุ่มค่าถัดไปเมื่อทราบค่าก่อนหน้า p ค่าได้เป็น

$$\hat{x}_t = -\sum_{i=1}^p \alpha_i x_{t-i} \dots\dots\dots (2.17)$$

จากสมการที่ (2.16) และ (2.17) จะได้ว่า

$$x_t - \hat{x}_t = \varepsilon_t \dots\dots\dots (2.18)$$

ดังนั้นจากสมการที่ (2.16) จะได้ว่า $\{\alpha_i\}$ เป็นสัมประสิทธิ์ของการประมาณพั้นระเชิงเส้นและ ε_t เป็นค่าความผิดพลาดตกค้างดังสมการที่ (2.18)

กำหนดคุณสมบัติของตัวกรองการประมาณพั้นระเชิงเส้นให้เป็น

$$F(z) = -\sum_{i=1}^p \alpha_i z^{-i} \dots\dots\dots (2.19)$$

กำหนดให้ $\hat{X}(z) \leftrightarrow \hat{x}_t$ และ $X(z) \leftrightarrow x_t$ เป็นคู่การแปลง Z จะได้ว่า

$$\hat{X}(z) = F(z) X(z) \dots\dots\dots (2.20)$$

ดังนั้นจากสมการที่ (2.17) และ (2.18) รูปแบบของการประมาณพหุระเชิงเส้นในรูปของการแปลง Z จะเขียนได้เป็น

$$X(z)(1 - F(z)) = E(z) \dots\dots\dots (2.21)$$

หรือ
$$X(z)A(z) = E(z) \dots\dots\dots (2.22)$$

เมื่อ
$$A(z) = 1 + \sum_{i=1}^p \alpha_i z^{-i} = 1 - F(z) \dots\dots\dots (2.23)$$

วิธีการประมาณค่าสัมประสิทธิ์ของการประมาณพหุระเชิงเส้น $\{\alpha_i\}$ สามารถกระทำได้โดยใช้วิธีการค่าความเพี้ยนกำลังสองเฉลี่ยน้อยที่สุด (Least Mean Square Error) กับสมการ (2.18) เมื่อพิจารณาในช่วง $[t_0, t_1]$ จะได้ค่าความเพี้ยนกำลังสองรวม β มีค่าเป็น

$$\begin{aligned} \beta &= \sum_{t=t_0}^{t_1} \varepsilon_t^2 = \sum_{t=t_0}^{t_1} \left(\sum_{i=0}^p \alpha_i x_{t-i} \right)^2 \\ &= \sum_{t=t_0}^{t_1} \sum_{i=0}^p \sum_{j=0}^p \alpha_i \alpha_j x_{t-i} x_{t-j} \end{aligned} \dots\dots\dots (2.24)$$

เมื่อ $\alpha_0 = 1$ กำหนดให้

$$c_{ij} = \sum_{t=t_0}^{t_1} x_{t-i} x_{t-j} \dots\dots\dots (2.25)$$

ดังนั้น β สามารถเขียนได้เป็น

$$\beta = \sum_{i=0}^p \sum_{j=0}^p \alpha_i c_{ij} \alpha_j \dots\dots\dots (2.26)$$

ทำการหาค่าต่ำสุดของ β ด้วยการหาอนุพันธ์ย่อยของ β เทียบกับ α_j ($j = 1, 2, \dots, p$) โดยกำหนดให้เป็นศูนย์ ดังนั้นจากสมการ (2.26) จะได้เป็น

$$\frac{\partial \beta}{\partial \alpha_j} = 2 \sum_{i=0}^p \alpha_i c_{ij} = 0 \quad (j = 1, 2, \dots, p) \dots\dots\dots (2.27)$$

ดังนั้นการหาค่าสัมประสิทธิ์ของการประมาณ $\{\alpha_i\}$ ก็คือการหาค่าตอบของชุดสมการเชิงเส้น p สมการ เมื่อทราบค่า c_{ij} ($i = 0, 1, 2, \dots, p; j = 1, 2, \dots, p$) ได้โดยหาจากสมการ (2.25) ซึ่งแสดงว่าค่า x_t จาก $t_0 - p$ ถึง t_1 มีนัยสำคัญ

ในการหาค่าตอบของลำดับเสียงพูด N ค่า สามารถหาได้ 2 วิธีคือ วิธีการแปรปรวนร่วม (Covariance Method) และวิธีการอัตโนมัติสัมพันธ์ (Autocorrelation Method) แต่ในที่นี้จะกล่าวถึงเพียงวิธีการอัตโนมัติสัมพันธ์เท่านั้น โดยวิธีการอัตโนมัติสัมพันธ์จะกำหนดให้เวลาอยู่ในช่วง $t_0 = -\infty$ และ $t_1 = \infty$ โดยกำหนดให้ $x_t = 0$ เมื่อ $t < 0$ และ $t \geq N$ จะได้ว่า

$$\begin{aligned} c_{ij} &= \sum_{t=-\infty}^{\infty} x_{t-i} x_{t-j} = \sum_{t=-\infty}^{\infty} x_t x_{t+|i-j|} \\ &= \sum_{t=0}^{N-1-|i-j|} x_t x_{t+|i-j|} = r_{|i-j|} \end{aligned} \dots\dots\dots (2.28)$$

ดังนั้นค่า α , จึงสามารถหาได้จาก

$$\sum_{i=0}^p \alpha_i r_{|i-j|} = 0 \quad (j = 1, 2, \dots, p) \dots\dots\dots (2.29)$$

เมื่อ

$$r_\tau = \sum_{i=0}^{N-1-\tau} x_i x_{i+\tau} \quad (\tau \geq 0) \dots\dots\dots (2.30)$$

โดยที่สมการ (2.30) สามารถเขียนในรูปสมการเมตริกซ์ได้ดังนี้

$$\begin{bmatrix} r_0 & r_1 & \dots & r_{p-1} \\ r_1 & r_0 & & \vdots \\ \vdots & & \ddots & r_1 \\ r_{p-1} & \dots & r_1 & r_0 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_p \end{bmatrix} \dots\dots\dots (2.31)$$

การหาค่าสัมประสิทธิ์ของการประมาณพหุระเชิงเส้น

จากการหาค่าสัมประสิทธิ์ของการประมาณพหุระเชิงเส้นโดยอาศัยออตสสัมพัทธ์นั้น เริ่มต้นจากกำหนดให้ E เป็นค่าพลังงานของความผิดพลาด เมื่อ $e(n)$ เป็นความผิดพลาดตกค้างที่สัมพันธ์กับสัญญาณ $x(n)$

$$\begin{aligned} E &= \sum_{-\infty}^{\infty} e^2(n) \dots\dots\dots (2.32) \\ &= \sum_{-\infty}^{\infty} \left[x(n) - \sum_{k=1}^p a_k x(n-k) \right]^2 \end{aligned}$$

การหาค่า a_k ที่ทำให้ E มีค่าต่ำที่สุดสามารถกระทำได้โดยการกำหนดให้ $\partial E / \partial a_k = 0$ สำหรับแต่ละ $k = 1, 2, 3, \dots, p$ จะได้สมการเชิงเส้น p สมการ ซึ่งก็คือ a_k ที่ไม่ทราบค่า p ตัวดังนี้

$$\sum_{-\infty}^{\infty} x(n-i)x(n) = \sum_{k=1}^p a_k \sum_{n=-\infty}^{\infty} x(n-i) \cdot x(n-k) \dots\dots\dots (2.33)$$

เนื่องจากพจน์แรกเป็นออตสสัมพัทธ์ $R(i)$ ของ $x(n)$ และ $x(n)$ มีความยาวจำกัดจะได้

$$\sum_{k=1}^p a_k R(i-k) = R(i), \quad 1 \leq i \leq p \dots\dots\dots (2.34)$$

เมื่อ

$$R(i) = \sum_{n=i}^{N-1} x(n)x(n-i) \dots\dots\dots (2.35)$$

เมื่อเขียนในรูปของสมการเมตริกซ์จะได้เป็น $\mathbf{R}\mathbf{a} = \mathbf{r}$ ตามสมการที่ (2.31) และเนื่องจากเมตริกซ์ \mathbf{R} เป็นชนิด Toeplitz เมตริกซ์ซึ่งมีค่าในแนวทแยงเท่ากันทั้งหมดทุกแนว ดังนั้นการคำนวณหาค่า a_k ในเมตริกซ์ \mathbf{a} จะอาศัยขั้นตอนวิธีการวนซ้ำของ Levinson-Durbin (O'Shaughnessy, 1988) เข้ามาช่วย

ขั้นตอนวิธีการวนซ้ำของ Levinson-Durbin

ขั้นตอนวิธีการนี้เป็นเทคนิคที่ช่วยในการคำนวณค่าสัมประสิทธิ์ของการประมาณพหุคูณเชิงเส้น a_m ในเมทริกซ์ \mathbf{a} เมื่อ $m = 1, 2, \dots, p$ และ p เป็นลำดับ (Order) ของการวิเคราะห์หาสัมประสิทธิ์ของการประมาณพหุคูณเชิงเส้น ซึ่งขั้นตอนวิธีการนี้จะให้คำตอบที่มีความเสถียรและแน่นอน โดยมีขั้นตอนแสดงในตารางที่ 2.1 ดังนี้

ตารางที่ 2.1 รายละเอียดขั้นตอนวิธีการวนซ้ำของ Levinson-Durbin

ขั้นตอนที่ 1 เงื่อนไขเริ่มต้น

$$\begin{aligned} E_0 &= R(0) \\ a_0 &= 0 \end{aligned} \quad \dots\dots\dots (2.36)$$

ขั้นตอนที่ 2 สัมประสิทธิ์การสะท้อน

$$k_m = \frac{R(m) - \sum_{i=1}^{m-1} a_{m-1}(i)R(m-i)}{E_{m-1}}, \quad |k_m| \leq 1 \quad \dots\dots\dots (2.37)$$

ขั้นตอนที่ 3 สัมประสิทธิ์ของการประมาณพหุคูณเชิงเส้นในแต่ละรอบของการคำนวณ

$$\begin{aligned} a_m(m) &= k_m \\ a_m(i) &= a_{m-1}(i) - k_m a_{m-1}(m-i), \quad (1 \leq i < m) \\ E_m &= (1 - k_m^2) E_{m-1} \end{aligned} \quad \dots\dots\dots (2.38)$$

ขั้นตอนที่ 4 สมการอัดสหสัมพันธ์

$$R(i) = \sum_{n=i}^{N-1} x(n)x(n-i), \quad i = 1, 2, \dots, p \quad \dots\dots\dots (2.39)$$

ขั้นตอนที่ 5 สัมประสิทธิ์ของการประมาณพหุคูณเชิงเส้นเมื่อสิ้นสุดการคำนวณ

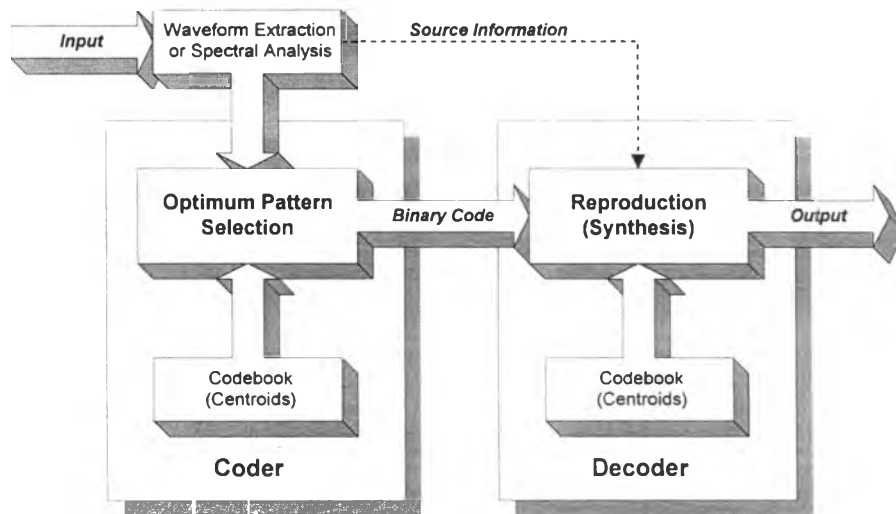
$$a_m = a_m(p), \quad 1 \leq m \leq p \quad \dots\dots\dots (2.40)$$

2. การควอนไทซ์แบบเวกเตอร์ (Vector Quantization)

การควอนไทซ์แบ่งได้เป็น 2 ประเภท (Makhoul, Roucos, and Gish, 1985) ได้แก่การควอนไทซ์แบบสเกลาร์และการควอนไทซ์แบบเวกเตอร์ การควอนไทซ์แบบสเกลาร์นั้นชุดของพารามิเตอร์หรือลำดับของสัญญาณจะถูกควอนไทซ์แยกจากกัน ส่วนการควอนไทซ์แบบเวกเตอร์นั้นชุดของพารามิเตอร์จะถูกควอน

ไทม์รวมกันเป็นเวกเตอร์เดียว จุดประสงค์ของการควอนไทซ์แบบเวกเตอร์ก็เพื่อลดขนาดจำนวนข้อมูลลง ซึ่งถือได้ว่าเป็นการบีบอัดข้อมูลวิธีการหนึ่ง เมื่อนำมาประยุกต์ใช้งานกับเสียงพูดจึงเรียกว่าเป็นการบีบอัดเสียงพูดหรือการเข้ารหัสเสียงพูดนั่นเอง

ในงานวิจัยนี้ได้นำเทคนิคการควอนไทซ์แบบเวกเตอร์มาใช้ในการควอนไทซ์ข้อมูลสัมประสิทธิ์ของการประมาณพัลลวะเชิงเส้นของเสียงพูด เพื่อลดขนาดของข้อมูลให้เหลือน้อยลงโดยมีความเพี้ยนน้อยที่สุด ด้วยการแทนที่เวกเตอร์ของสัมประสิทธิ์ของการประมาณพัลลวะเชิงเส้นจากสัญญาณเสียงพูดซึ่งตัดออกมาพิจารณาด้วยเวกเตอร์ค่าหนึ่งทีใกล้เคียงมากที่สุด โดยอาศัยการวัดความเพี้ยนระหว่างเวกเตอร์ทั้งสองให้มค่าน้อยที่สุด



รูปที่ 2.6 รายละเอียดขั้นตอนการควอนไทซ์แบบเวกเตอร์ (Sadaoki Furui, 1989)

รายละเอียดขั้นตอนการควอนไทซ์แบบเวกเตอร์ดังแสดงในรูปที่ 2.6 เริ่มต้นจากการที่สัญญาณเสียงได้รับการเปรียบเทียบกับชุดรหัสที่มีอยู่ โดยการคำนวณหาค่าความเพี้ยนแล้วพิจารณารหัสที่ให้ค่าความเพี้ยนน้อยที่สุด สัญญาณเสียงดังกล่าวจะถูกแทนที่ด้วยชุดรหัสนั้น

ทฤษฎีของขั้นตอนวิธีการควอนไทซ์แบบเวกเตอร์ เริ่มจากสมมติให้เวกเตอร์ขนาด N มิติ $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_N]^T$ ซึ่งมีส่วนประกอบ $\{x_k, 1 \leq k \leq N\}$ เป็นตัวแปรสุ่มซึ่งมีแอมพลิจูดต่อเนื่องและเป็นจำนวนจริง ในการควอนไทซ์แบบเวกเตอร์นั้นเวกเตอร์ \mathbf{x} จะถูกจับคู่ให้ตรงกับเวกเตอร์ \mathbf{y} ซึ่งมีแอมพลิจูดไม่ต่อเนื่องและเป็นจำนวนจริง หรือเรียกได้ว่า \mathbf{x} ถูกควอนไทซ์เป็น \mathbf{y} และ \mathbf{y} เป็นค่าควอนไทซ์ของ \mathbf{x} นั่นคือ

$$\mathbf{y} = q(\mathbf{x}) \dots\dots\dots (2.41)$$

เมื่อ $q(\cdot)$ เป็นตัวดำเนินการควอนไทซ์ นอกจากนี้ y ยังอาจเรียกได้ว่า "เวกเตอร์สร้างใหม่" หรือ "เวกเตอร์ขาออก" ที่สัมพันธ์กับ x โดยทั่วไปแล้ว y เป็นเพียงหนึ่งในชุดของ $Y = \{y_i, 1 \leq i \leq L\}$ เมื่อ $y_i = [y_{i1} \ y_{i2} \ \dots \ y_{iM}]^T$ ซึ่งชุด Y เป็นชุดรหัสที่สร้างใหม่หรือชุดรหัส L เป็นขนาดของชุดรหัส และ y_i เป็นชุดของเวกเตอร์รหัส ในด้านการรู้จำรูปแบบนั้น y_i ถือว่าเป็นรูปแบบอ้างอิงหรือชุดรูปร่างต้นแบบ ขนาด L ของชุดรหัสเรียกว่า "จำนวนชั้น" ดังนั้นการออกแบบชุดรหัสจะใช้การแบ่งปริภูมิ N มิติของเวกเตอร์สุ่ม x ออกเป็น L ภาคหรือ "เซลล์" $\{C_i, 1 \leq i \leq L\}$ และกำหนดความสัมพันธ์ระหว่างแต่ละเซลล์ C_i กับเวกเตอร์ y_i ดังนั้นการควอนไทซ์จะเป็นการกำหนดเวกเตอร์รหัส y_i ถ้า x อยู่ใน C_i ตามเงื่อนไขในสมการที่ (2.42)

$$q(x) = y_i, \quad \text{if } x \in C_i, \dots \dots \dots (2.42)$$

ขั้นตอนการออกแบบชุดรหัสตามเงื่อนไขนี้เรียกว่า "การฝึกฝน" หรือ "การสร้างชุดรหัส" เมื่อ x ถูกควอนไทซ์เป็น y จะเกิดความเพี้ยนจากการควอนไทซ์ขึ้น โดยการวัดค่าความเพี้ยน $d(x, y)$ ระหว่าง x และ y ซึ่ง $d(x, y)$ ก็คือการวัดความไม่คล้ายคลึงกันหรือการวัดระยะห่าง ดังนั้นความเพี้ยนเฉลี่ยโดยรวมแสดงได้ดังนี้

$$D = \lim_{M \rightarrow \infty} \frac{1}{M} \sum_{n=1}^M d[x(n), y(n)] \dots \dots \dots (2.43)$$

ถ้า $x(n)$ ไม่แปรเปลี่ยนตามเวลาและมีความเป็นเออร์กอดิก ดังนั้นความเพี้ยนเฉลี่ยโดยรวมจะถูกจำกัดไว้ดังนี้

$$\begin{aligned} D &= E[d(x, y)] \\ &= \sum_{i=1}^L P(x \in C_i) E[d(x, y_i) | x \in C_i] \dots \dots \dots (2.44) \\ &= \sum_{i=1}^L P(x \in C_i) \int_{x \in C_i} d(x, y_i) p(x) dx \end{aligned}$$

เมื่อ $P(x \in C_i)$ เป็นความน่าจะเป็นแบบไม่ต่อเนื่องที่ x อยู่ใน C_i ส่วน $p(x)$ เป็นฟังก์ชันความหนาแน่นความน่าจะเป็นแบบหลายมิติของ x และอินทิกรัลจะครอบคลุมส่วนประกอบของเวกเตอร์ x ทั้งหมด

2.1 การวัดค่าความเพี้ยน

การวัดค่าความเพี้ยนนั้นเป็นส่วนที่มีความสำคัญมากที่สุดส่วนหนึ่งในระบบการรู้จำคำพูด โดยถูกนำมาใช้ในขั้นตอนการสร้างและฝึกฝนชุดรหัสรวมทั้งในขั้นตอนวิธีการควอนไทซ์แบบเวกเตอร์ เนื่องจากความแตกต่างของค่าความเพี้ยนจะเป็นตัวบ่งชี้ถึงความเหมือนหรือความแตกต่างในคุณภาพของเสียงพูด วิธีการวัดค่าความเพี้ยนมีหลายวิธีได้แก่ ค่าความผิดพลาดกำลังสองเฉลี่ย (Mean-Square Error, MSE) ค่าความผิดพลาดกำลังสองเฉลี่ยถ่วงน้ำหนัก (Weighted Mean-Square Error) การวัดค่าความเพี้ยนโดย

อาศัยการประมาณพันระเชิงเส้น (Linear Prediction Distortion Measures) เป็นต้น (Makhoul, 1985) ดังมีรายละเอียดของแต่ละวิธีการดังนี้

1) ค่าความผิดพลาดกำลังสองเฉลี่ย (Mean-Square Error, MSE)

การวัดค่าความเพี้ยนโดยอาศัยค่าความผิดพลาดกำลังสองเฉลี่ย ถือว่าเป็นวิธีการวัดค่าที่ใช้กันมากที่สุด เนื่องจากความง่ายทั้งในการประยุกต์ใช้งานและในทางคณิตศาสตร์ โดยมีรูปแบบดังนี้

$$d_2(\mathbf{x}, \mathbf{y}) = \frac{1}{N}(\mathbf{x} - \mathbf{y})^T(\mathbf{x} - \mathbf{y}) = \frac{1}{N} \sum_{k=1}^N (x_k - y_k)^2 \dots\dots\dots (2.45)$$

หน่วยของการวัดค่าความเพี้ยนกำหนดให้มีหน่วยต่อมิติ กรณีทั่วไปของการวัดค่าความเพี้ยนในกรณีนี้บนพื้นฐานของ L_r Norm ดังนี้

$$d_r(\mathbf{x}, \mathbf{y}) = \frac{1}{N} \sum_{k=1}^N |x_k - y_k|^r \dots\dots\dots (2.46)$$

จากกรณีทั่วไปของการวัดค่าความเพี้ยนตามสมการที่ (2.36) นั้นจะสมมูลกับสมการที่ (2.45) ได้ในกรณีที่ $r = 2$ นอกจากนี้ยังมีค่าที่นิยมใช้กันมากอีก 2 ค่าได้แก่ค่า $r = 1$ และ $r = \infty$ โดยที่ d_1 แทนค่าความผิดพลาดสัมบูรณ์เฉลี่ย (Average Absolute Error) และ d_∞ มีแนวโน้มที่จะเข้าสู่ค่าความผิดพลาดสูงสุด ซึ่งแสดงว่า

$$\lim_{r \rightarrow \infty} [d_r(\mathbf{x}, \mathbf{y})]^{1/r} = \max \{|x_k - y_k|, 1 \leq k \leq N\} \dots\dots\dots (2.47)$$

การทำให้ D ที่ $r = \infty$ มีค่าน้อยที่สุด จะเทียบเท่ากับการทำให้ค่าความผิดพลาดที่มากที่สุดในการควอนไทซ์มีค่าน้อยที่สุด ในการเข้ารหัสเสียงพูดนั้น d_2 เป็นวิธีการวัดค่าความเพี้ยนที่ถูกนำมาใช้มากที่สุด ส่วน d_1 และ d_∞ ก็ถูกนำมาใช้บ้างเป็นครั้งคราว

2) ค่าความผิดพลาดกำลังสองเฉลี่ยถ่วงน้ำหนัก (Weighted Mean Square Error)

จากค่าความผิดพลาดกำลังสองเฉลี่ย d_2 นั้นกำหนดให้ค่าความเพี้ยนที่เกิดขึ้นโดยการควอนไทซ์ค่าพารามิเตอร์ $\{x_k\}$ ต่างกันมีน้ำหนักสมมูลย์ โดยทั่วไปแล้วน้ำหนักที่ไม่สมมูลย์สามารถกำหนดขึ้นได้โดยการให้ความสำคัญกับค่าความเพี้ยนค่าหนึ่งมากกว่าค่าอื่นๆ กรณีทั่วไปของค่าความผิดพลาดกำลังสองเฉลี่ยถ่วงน้ำหนักเป็นดังนี้

$$d_w(\mathbf{x}, \mathbf{y}) = (\mathbf{x} - \mathbf{y})^T \mathbf{W}(\mathbf{x} - \mathbf{y}) \dots\dots\dots (2.48)$$

เมื่อ \mathbf{W} เป็นเมตริกซ์ถ่วงน้ำหนักที่เป็นบวกทั้งหมดเสมอ และถ้า $\mathbf{W} = N^{-1}\mathbf{I}$ เมื่อ \mathbf{I} เป็นเมตริกซ์เอกลักษณ์ ผลลัพธ์ที่ได้ก็คือ $d_w = d_2$ ตัวเลือกหนึ่งของ \mathbf{W} ที่นำมาประยุกต์ใช้กับการจำแนกรูปแบบก็คือ $\mathbf{W} = \Gamma^{-1}$ เมื่อ Γ เป็นเมตริกซ์ของความแปรปรวนร่วมของเวกเตอร์สุ่ม \mathbf{x}

$$\Gamma = E[(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{x} - \bar{\mathbf{x}})^T], \quad \mathbf{x} = E[\bar{\mathbf{x}}] \dots\dots\dots (2.49)$$

ในกรณีนี้ทำให้ d_w ลดรูปลงอยู่ในรูปของระยะทาง Mahalanobis ดังนี้

$$d_w(\mathbf{x}, \mathbf{y}) = (\mathbf{x} - \mathbf{y})^T \Gamma^{-1} (\mathbf{x} - \mathbf{y}) \dots\dots\dots (2.50)$$

ถ้าเมตริกซ์ \mathbf{W} มีความสมมาตรนอกเหนือจากการเป็นบวกทั้งหมดเสมอ ดังนั้น

$$\mathbf{W} = \mathbf{P}^T \mathbf{P} \dots\dots\dots (2.51)$$

เวกเตอร์ \mathbf{x} และ \mathbf{y} สามารถแปลงไปเป็นชุดเวกเตอร์ $\tilde{\mathbf{x}}$ และ $\tilde{\mathbf{y}}$ ดังนี้

$$\tilde{\mathbf{x}} = \mathbf{P}\mathbf{x} \quad \tilde{\mathbf{y}} = \mathbf{P}\mathbf{y} \dots\dots\dots (2.52)$$

$$\begin{aligned} d_w(\mathbf{x}, \mathbf{y}) &= (\mathbf{P}\mathbf{x} - \mathbf{P}\mathbf{y})^T (\mathbf{P}\mathbf{x} - \mathbf{P}\mathbf{y}) \\ &= (\tilde{\mathbf{x}} - \tilde{\mathbf{y}})^T (\tilde{\mathbf{x}} - \tilde{\mathbf{y}}) \dots\dots\dots (2.53) \\ &= d_2(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \end{aligned}$$

ซึ่งแสดงว่าค่าความผิดพลาดกำลังสองเฉลี่ยถ่วงน้ำหนักระหว่างค่าเวกเตอร์ต้นฉบับจะมีค่าเท่ากับค่าความผิดพลาดกำลังสองเฉลี่ยระหว่างเวกเตอร์ที่เปลี่ยนรูปแล้ว ดังนั้นจึงควรเปลี่ยนรูปเวกเตอร์ตามสมการที่ (2.42) กับข้อมูลทั้งหมดก่อนทำการควอนไทซ์แบบเวกเตอร์

3) การวัดค่าความเพี้ยนโดยอาศัยการประมาณพันธะเชิงเส้น

(Linear Prediction Distortion Measures)

ในการวิเคราะห์หาสัมประสิทธิ์ของการประมาณพันธะเชิงเส้นนั้น สัมประสิทธิ์ของการประมาณ $\{a(k)\}$ หาได้จากผลลัพธ์ของการทำให้ค่าพลังงานจากค่าคงเหลือของการประมาณมีค่าต่ำที่สุด ซึ่งก็คือผลลัพธ์ของชุดสมการเชิงเส้นดังนี้

$$\sum_{k=1}^N a(k)\phi(i-k) = -\phi(i), \quad 1 \leq i \leq N \dots\dots\dots (2.54)$$

เมื่อ $\{\phi(i), 0 \leq i \leq N\}$ เป็นสัมประสิทธิ์ของอัตสหสัมพันธ์ช่วงสั้นของสัญญาณเสียงในกรอบเดียว ดังนั้นอัตราขยาย G ของตัวกรอง $H(z)$ จะถูกกำหนดขึ้นโดยให้พลังงานขาออกมีค่าเท่ากับ $\phi(0)$ เมื่อถูกกระตุ้นด้วยแหล่งที่มีความแปรปรวนหนึ่งหน่วยดังนี้

$$G^2 = \phi(0) + \sum_{k=1}^N a(k)\phi(k) \dots\dots\dots (2.55)$$

ซึ่งมีค่าเท่ากับพลังงานคงเหลือที่มีค่าต่ำที่สุด และเนื่องจากความไม่เสถียรของตัวกรองที่เป็นโพลทั้งหมด จึงต้องมีการแปลงไปเป็นพารามิเตอร์ชุดใหม่เรียกว่า "สัมประสิทธิ์การสะท้อน" $\{K_k, 1 \leq k \leq N\}$ หรือ "สัมประสิทธิ์ของอัตสัมพันธ์ส่วนย่อย" ซึ่งสามารถหาได้จากการหาผลลัพธ์ของสมการที่ (2.54) หรือหากจากค่าสัมประสิทธิ์ของการประมาณ สำหรับ $H(z)$ ที่เสถียรนั้นสัมประสิทธิ์การสะท้อนจะต้องมีคุณสมบัติดังนี้

$$|K_k| < 1, \quad 1 \leq k \leq N \dots\dots\dots (2.56)$$



เนื่องจากเมื่อค่าของ $|K_k|$ มีค่าเข้าใกล้ 1 จะทำให้โพลเข้าใกล้วงกลมหนึ่งหน่วยมากขึ้น ดังนั้นเมื่อ K_k เปลี่ยนไปเพียงเล็กน้อยจะส่งผลให้สเปกตรัมเปลี่ยนแปลงไปมาก ดังนั้นจึงต้องแปลงไปเป็น พารามิเตอร์ชุดใหม่ดังนี้

$$S_k = \frac{2}{\pi} \sin^{-1} K_k, \quad 1 \leq k \leq N \dots\dots\dots (2.57)$$

$$G_k = \frac{1}{2} \log \frac{1-K_k}{1+K_k} = \tanh^{-1} K_k, \quad 1 \leq k \leq N \dots\dots\dots (2.58)$$

เมื่อ G_k เป็นอัตราส่วนพื้นที่ล็อก (Log-Area-Ratios, LARs) ซึ่งมีสัดส่วนสัมพันธ์กับการเปลี่ยนแปลงของล็อกสเปกตรัมของ $H(z)$ โดยใช้ค่าความผิดพลาดกำลังสองเฉลี่ย d_2 และค่าความผิดพลาดสูงสุดที่ต่ำที่สุด d_∞ ในการควอนไทซ์ S_k และ G_k

4) การวัดค่าความเพี้ยนของ Itakura-Saito (Itakura-Saito Distortion Measures)

เทคนิคการวัดค่าความเพี้ยนในการควอนไทซ์สัมประสิทธิ์การประมาณพหุของ Itakura-Saito นี้อยู่บนพื้นฐานของหลักการความน่าจะเป็นจริงสูงสุด (Maximum-Likelihood Principles) โดยอาศัยหลักการของการวัดค่าความเพี้ยนระหว่างเวกเตอร์ของสัมประสิทธิ์การประมาณพหุ $\mathbf{x} = [a(1) \ a(2) \ \dots \ a(N)]^T$ กับเวกเตอร์ของสัมประสิทธิ์การประมาณพหุ \mathbf{y} ดังนี้

$$d_I(\mathbf{x}, \mathbf{y}) = (\mathbf{x} - \mathbf{y})^T \Phi_x (\mathbf{x} - \mathbf{y}) \dots\dots\dots (2.59)$$

$$\Phi_x = \{\phi(i-k)/\phi(0), 0 \leq i, k \leq N-1\} \dots\dots\dots (2.60)$$

เมื่อ Φ_x เป็นเมตริกซ์ออสทสสัมพันธ์ที่ได้รับการปรับให้เป็นบรรทัดฐานเดียวกัน โดยมีสัมประสิทธิ์ $\phi(i-k)$ ที่ใช้ในการคำนวณหาค่าเวกเตอร์ของสัมประสิทธิ์การประมาณพหุ \mathbf{x} ตามสมการที่ (2.54) เนื่องจากค่าสัมประสิทธิ์ของออสทสสัมพันธ์ในสมการที่ (2.60) ได้รับการปรับให้เป็นบรรทัดฐานเดียวกันด้วยค่า $\phi(0)$ จึงแสดงว่าเมตริกซ์ Φ_x และเวกเตอร์ \mathbf{x} ใช้ในการหาค่าซึ่งกันและกัน นอกจากนี้ Φ_x ในสมการที่ (2.59) เป็นเมตริกซ์ดวงน้ำหนักที่มีการเปลี่ยนแปลงค่าเมื่อค่าของ \mathbf{x} เปลี่ยนแปลงไป แต่ในสมการที่ (2.48) นั้นค่า \mathbf{W} มีค่าคงที่เสมอ ดังนั้นการหาค่าความเพี้ยนด้วยเทคนิคนี้ไม่มีความสมมาตรอันเนื่องมาจากค่าเมตริกซ์ $\Phi_x \neq \Phi_y$ เมื่อ $\mathbf{x} \neq \mathbf{y}$ เช่น $d_I(\mathbf{x}, \mathbf{y}) \neq d_I(\mathbf{y}, \mathbf{x})$ เป็นต้น

2.2 การออกแบบชุดรหัส

การออกแบบชุดรหัส L ชั้นนั้น อาศัยวิธีการแบ่งปริภูมิ N มิติออกเป็น L เซลล์ $\{C_i, 1 \leq i \leq L\}$ และกำหนดให้แต่ละเซลล์ C_i สัมพันธ์กับเวกเตอร์ \mathbf{y}_i โดยตัวควอนไทซ์จะกำหนดเวกเตอร์รหัส \mathbf{y}_i ถ้า \mathbf{x} อยู่ใน C_i ซึ่งจะมีความเหมาะสมที่สุดหรือมีความเพี้ยนน้อยที่สุดเมื่อค่าความเพี้ยนตามสมการที่ (2.44) มีค่าน้อยที่สุดจากตัวควอนไทซ์ L ชั้นทั้งหมด โดยมีเงื่อนไขของความเหมาะสม 2

ประการดังนี้ เงื่อนไขแรก ตัวควอนไทล์ที่เหมาะสมที่สุดต้องเป็นไปตามกฎความเพี้ยนน้อยที่สุดหรือกฎการเลือกบริเวณที่ใกล้เคียงที่สุด (Nearest Neighbour Rule) โดยตัวควอนไทล์จะเลือกเวกเตอร์รหัสที่มีความเพี้ยนน้อยที่สุดเมื่อเทียบกับ \mathbf{x} ดังนี้

$$q(\mathbf{x}) = \mathbf{y}_i, \text{ iff } d(\mathbf{x}, \mathbf{y}_i) \leq d(\mathbf{x}, \mathbf{y}_j), j \neq i, 1 \leq j \leq L \dots\dots\dots (2.61)$$

เงื่อนไขที่สอง การเลือกเวกเตอร์รหัส \mathbf{y}_i จะต้องให้ความเพี้ยนเฉลี่ยในเซลล์ C_i มีค่าน้อยที่สุด นั่นคือ \mathbf{y}_i เป็นเวกเตอร์ \mathbf{y} ที่ทำให้สมการที่ (2.62) มีค่าน้อยที่สุด

$$D_i = E[d(\mathbf{x}, \mathbf{y}) | \mathbf{x} \in C_i] = \int_{\mathbf{x} \in C_i} d(\mathbf{x}, \mathbf{y}) p(\mathbf{x}) d\mathbf{x} \dots\dots\dots (2.62)$$

เวกเตอร์ดังกล่าวนี้เรียกว่า "จุดศูนย์กลาง" (Centroid) ของเซลล์ C_i ซึ่งเขียนได้เป็น

$$\mathbf{y}_i = \text{cent}(C_i) \dots\dots\dots (2.63)$$

การหาค่าจุดศูนย์กลางของแต่ละบริเวณนั้นจะขึ้นอยู่กับนิยามของการวัดค่าความเพี้ยน ในทางปฏิบัติจะกำหนดชุดของเวกเตอร์ฝึกฝน $\{\mathbf{x}(n), 1 \leq n \leq M\}$ โดยที่ชุดย่อย M_i ของเวกเตอร์นี้อยู่ในเซลล์ C_i ซึ่งมีความเพี้ยนเฉลี่ย D_i ดังนี้

$$D_i = \frac{1}{M_i} \sum_{\mathbf{x} \in C_i} d(\mathbf{x}, \mathbf{y}_i) \dots\dots\dots (2.64)$$

เมื่อพิจารณาในกรณีของค่าความผิดพลาดกำลังสองเฉลี่ยและค่าความผิดพลาดกำลังสองเฉลี่ยถ่วงน้ำหนัก โดยที่ \mathbf{y}_i เป็นเพียงค่าเฉลี่ยสุ่มของเวกเตอร์ฝึกฝนทั้งหมดที่อยู่ภายใน C_i ดังนั้นความเพี้ยนเฉลี่ย D_i จะลดรูปลงเหลือเพียง

$$\mathbf{y}_i = \frac{1}{M_i} \sum_{\mathbf{x} \in C_i} \mathbf{x}(n) \dots\dots\dots (2.65)$$

เมื่อพิจารณาในกรณีค่าความเพี้ยนของ Itakura-Saito d_i นั้น การหาค่า \mathbf{y}_i จะอาศัยการเฉลี่ยค่าอัตสหสัมพันธ์ที่ถูกทำให้เป็นบรรทัดฐานเดียวกันและสัมพันธ์กับเวกเตอร์สุ่ม โดยที่ $\phi_x(k)$ ได้รับการทำให้เป็นบรรทัดฐานเดียวกันทำให้ $\phi_x(0) = 1$ ดังนั้นเวกเตอร์สามารถหาได้จากคำตอบของสมการที่ (2.44) โดยใช้ $\phi_y(k)$ เป็นสัมประสิทธิ์ของอัตสหสัมพันธ์

$$\phi_y(k) = \frac{1}{M_i} \sum_{\mathbf{x} \in C_i} \phi_x(k), \quad 0 \leq k \leq N \dots\dots\dots (2.66)$$

นอกจากนี้ยังมีอีกวิธีการหนึ่งในการออกแบบชุดรหัสก็คือขั้นตอนวิธีการแบ่งกลุ่มแบบวนซ้ำ (Iterative Clustering Algorithm) หรือขั้นตอนวิธีการแบ่งเฉลี่ย K ส่วน (K-Means Algorithm) เมื่อกำหนดให้ $K = L$ จะเป็นการแบ่งชุดของเวกเตอร์ฝึกฝน $\{\mathbf{x}(n)\}$ ออกเป็น L กลุ่ม C_i โดยเป็นไปตามเงื่อนไขของความเหมาะสมที่จำเป็นทั้งสองประการ ขั้นตอนวิธีการนี้กำหนดให้ m เป็นดัชนีของการวนซ้ำและ $C_i(m)$ เป็นกลุ่มที่ i ในรอบที่ m โดยมี $\mathbf{y}_i(m)$ เป็นจุดศูนย์กลางดังแสดงในตารางที่ 2.1

เนื่องจากขั้นตอนวิธีการแบ่งเฉลี่ย K ส่วนนี้จะลู่เข้าสู่ค่าเหมาะสมที่สุดเฉพาะแห่ง (Local Optimum) จึงทำให้ผลลัพธ์ที่ได้ไม่เป็นหนึ่งเดียว ดังนั้นการหาค่าเหมาะสมที่สุดที่ครอบคลุมทั้งหมด

(Global Optimum) จะกระทำได้โดยการใช้ค่าเริ่มต้นของเวกเตอร์รหัสหลายค่าที่แตกต่างกันไป จากนั้นจึงทำซ้ำขั้นตอนวิธีการดังกล่าวนี้กับชุดของค่าเริ่มต้นหลายชุดที่แตกต่างกัน แล้วจึงเลือกชุดรหัสที่ให้ค่าความเพี้ยนทั้งหมดน้อยที่สุด

ตารางที่ 2.2 รายละเอียดขั้นตอนวิธีการแบ่งเฉลี่ย K ส่วน

ขั้นตอนที่ 1	กำหนดให้ $m = 0$ และเลือกชุดของเวกเตอร์รหัสเริ่มต้น
กระบวนการเริ่มต้น	$y_i(0), 1 \leq i \leq L$ ด้วยวิธีการที่เหมาะสม
ขั้นตอนที่ 2	ทำการจำแนกชุดของเวกเตอร์ฝึกฝน $\{x(n), 1 \leq n \leq M\}$ ไปตามกลุ่ม
กระบวนการจำแนก	ด้วยกฎการเลือกเพื่อนบ้านใกล้เคียงที่สุดดังนี้ $x \in C_i(m), \text{ iff } d[x, y_i(m)] \leq d[x, y_j(m)], \text{ all } j \neq i$
ขั้นตอนที่ 3	กำหนดให้ $m \leftarrow m+1$ เพื่อปรับค่าของเวกเตอร์รหัสของทุกกลุ่มโดย
กระบวนการปรับค่า	การคำนวณจุดศูนย์กลางของเวกเตอร์รหัสในแต่ละกลุ่มใหม่ดังนี้
เวกเตอร์รหัส	$y_i(m) = \text{cent}(C_i(m)), 1 \leq i \leq L$
ขั้นตอนที่ 4	ถ้าค่าความเพี้ยนโดยรวม $D(m)$ ของการวนซ้ำรอบที่ m เมื่อเทียบกับ
กระบวนการสิ้นสุด	$D(m-1)$ มีค่าลดลงต่ำกว่าจุดเริ่มเปลี่ยนให้ถือว่าสิ้นสุดกระบวนการ ถ้ามีค่าสูงกว่าให้ทำซ้ำขั้นตอนที่ 2 ใหม่

การควอนไทซ์แบบเวกเตอร์นั้นมีข้อได้เปรียบทางด้านประสิทธิภาพเหนือการควอนไทซ์แบบสเกลาร์โดยเฉพาะกับแหล่งข้อมูลที่ขึ้นแก่กันอย่างไม่เป็นเชิงเส้น ดังนั้นในการออกแบบสร้างชุดรหัสที่มีประสิทธิภาพจะต้องอาศัยวิธีการที่เหมาะสม ในการเข้ารหัสโดยการค้นหาทั่วทั้งหมดนั้นค่าใช้จ่ายจะเพิ่มขึ้นแบบเอกซ์โพเนนเชียลตามจำนวนบิตต่อเวกเตอร์ ค่าใช้จ่ายในการคำนวณและการจัดเก็บจะเพิ่มขึ้นเป็นสองเท่าต่ออัตราเร็วที่เพิ่มขึ้นแต่ละบิต

การควอนไทซ์แบบเวกเตอร์นั้นมีข้อได้เปรียบทางด้านประสิทธิภาพเหนือการควอนไทซ์แบบสเกลาร์โดยเฉพาะกับแหล่งข้อมูลที่ขึ้นแก่กันอย่างไม่เป็นเชิงเส้น ดังนั้นในการออกแบบสร้างชุดรหัสที่มีประสิทธิภาพจะต้องอาศัยวิธีการที่เหมาะสม ในการเข้ารหัสโดยการค้นหาทั่วทั้งหมดนั้นค่าใช้จ่ายจะเพิ่มขึ้นแบบเอกซ์โพเนนเชียลตามจำนวนบิตต่อเวกเตอร์ ค่าใช้จ่ายในการคำนวณและการจัดเก็บจะเพิ่มขึ้นเป็นสองเท่าต่ออัตราเร็วที่เพิ่มขึ้นแต่ละบิต

ขั้นตอนวิธีการค้นหาอย่างรวดเร็วที่ได้รับการนำเสนอในการรู้จำรูปแบบนั้นมีด้วยกันหลายวิธี ซึ่งได้รับการนำมาประยุกต์ใช้กับการควอนไทซ์แบบเวกเตอร์เพื่อลดการคำนวณที่เกิดจากการค้นหาหมดทั้งชุดรหัส ขั้นตอนวิธีการส่วนใหญ่จะอยู่บนพื้นฐานของหลักการเชิงเรขาคณิตในปริภูมิชนิด Euclidean ซึ่งต้องการกรรมวิธีประมวลผลเบื้องต้นกับชุดรหัส และเป็นการประนีประนอมกันระหว่างการคูณกับการเปรียบเทียบและความต้องการเนื้อที่จัดเก็บที่เพิ่มสูงขึ้น โดยจำนวนการคูณสามารถลดลงได้เป็นจำนวนเท่าของจำนวนเดิม

ขั้นตอนวิธีการออกแบบสร้างและฝึกฝนชุดรหัสซึ่งเป็นขั้นตอนหนึ่งในการสร้างชุดรูป้องต้นแบบของคำศัพท์แต่ละคำ จัดเป็นขั้นตอนการฝึกฝนชุดรหัสเพื่อใช้ในการควอนไทซ์แบบเวกเตอร์ ขั้นตอนวิธีการออกแบบสร้างชุดรหัสมีหลายวิธี (Makhoul, Roucos, and Gish, 1985) ได้แก่ การค้นหาแบบทวิภาค (Binary Search) การควอนไทซ์แบบต่อเรียงกัน (Cascade Quantization) รหัสผลคูณ (Product Codes) และชุดรหัสสุ่ม (Random Codebooks) สำหรับงานวิจัยนี้จะใช้วิธีการชุดรหัสสุ่มในการสร้างชุดรหัสเริ่มต้นเพื่อการฝึกฝนชุดรหัส เนื่องจากมีความรวดเร็วและมีประสิทธิภาพใกล้เคียงกับวิธีการค้นหาแบบทวิภาค ดังนั้นในที่นี้จะกล่าวถึงเพียงรายละเอียดของวิธีการค้นหาแบบทวิภาคและชุดรหัสสุ่มเท่านั้น

1) การค้นหาแบบทวิภาค (Binary Search)

เนื่องจากขั้นตอนวิธีการแบ่งเฉลี่ย K ส่วนโดยมี $K = L$ ชั้นนั้น จำเป็นต้องอาศัยการค้นหาทั่วทั้งหมัดใน L เวกเตอร์รหัสเพื่อควอนไทซ์แต่ละเวกเตอร์ที่รับเข้ามา ดังนั้นการค้นหาแบบทวิภาคหรือการแบ่งกลุ่มเป็นลำดับชั้น (Hierarchical Clustering) จึงเป็นวิธีการสำหรับแบ่งปริภูมิที่ทำให้การค้นหาเวกเตอร์รหัสที่มีค่าความเพี้ยนน้อยที่สุดเป็นส่วนกับ $\log_2 L$ แทนที่จะเป็น L โดยมีขั้นตอนคือ ในขั้นแรกปริภูมิ N มิติจะได้รับการแบ่งออกเป็น 2 บริเวณซึ่งใช้ขั้นตอนวิธีการแบ่งเฉลี่ย K เมื่อ $K = 2$ จากนั้นทั้งสองบริเวณก็จะถูกแบ่งออกเป็นสองบริเวณต่อไปเรื่อยๆ จนกระทั่งปริภูมิถูกแบ่งออกเป็น L บริเวณหรือเซลล์ จึงทำให้ค่าของ L จะต้องเป็นกำลังของ 2 เท่านั้น นั่นคือ $L = 2^B$ เมื่อ B เป็นจำนวนบิตทั้งหมด โดยมีจุดศูนย์กลางที่สัมพันธ์กับแต่ละบริเวณในการแบ่งแบบทวิภาคแต่ละครั้ง

แผนภูมิการแบ่งแบบทวิภาคของปริภูมิออกเป็น $L = 8$ เซลล์ดังแสดงในรูปที่ 2.3 ในการแบ่งแบบทวิภาคครั้งแรกจะได้ v_1 และ v_2 เป็นจุดศูนย์กลาง ในการแบ่งแบบทวิภาคครั้งที่สองประกอบด้วย 4 บริเวณซึ่งมีจุดศูนย์กลางเป็น v_3 ถึง v_6 จุดศูนย์กลางของบริเวณภายหลังการแบ่งแบบทวิภาคครั้งที่สามจะเป็นเวกเตอร์รหัส y_i ดังนั้นเวกเตอร์ขาเข้า v_i จะได้รับการควอนไทซ์โดยการไล่ค้นหาตามแผนภูมิต้นไม้ดังรูปที่ 2.2 ไปตามแต่ละปมบนเส้นทาง โดยจะเปรียบเทียบ x กับ v_1 และ v_2 ถ้า $d(x, v_2) < d(x, v_1)$ จึงเลือกเส้นทางไปยัง v_2 จากนั้นจะเปรียบเทียบ x กับ v_5 และ v_6 ถ้า $d(x, v_5) < d(x, v_6)$ จึงเลือกเส้นทางไปยัง v_5 และในขั้นตอนสุดท้ายจะเปรียบเทียบ x กับ y_5 และ y_6 ถ้า $d(x, y_6) < d(x, y_5)$ ดังนั้น y_6 จึงเป็นค่าควอนไทซ์ของ x

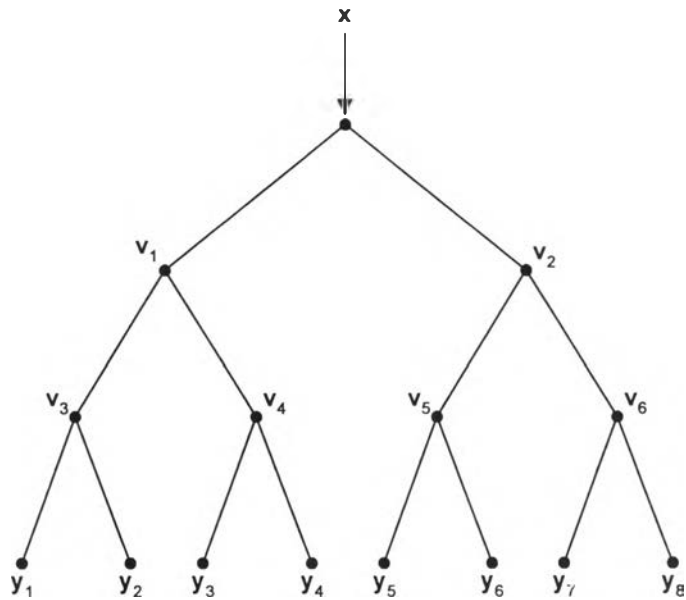
จากขั้นตอนวิธีการแบ่งแบบทวิภาค จำนวนการคำนวณค่าความเพี้ยนจะมีค่าเท่ากับ $2 \log_2 L$ เมื่อกำหนดให้การคำนวณค่าความเพี้ยนแต่ละครั้งมีการคูณและบวก N ครั้งจะมีค่าใช้จ่ายรวมทั้งหมดในการคำนวณเป็น

$$C = 2N \log 2L = 2NB \dots\dots\dots (2.67)$$

โดยมีความสัมพันธ์เป็นเชิงเส้นกับจำนวนบิต ซึ่งเป็นผลให้มีการลดจำนวนการคำนวณไปอย่างมหาศาล แต่ค่าใช้จ่ายในการจัดเก็บกลับเพิ่มสูงขึ้น นอกจากการจัดเก็บเวกเตอร์รหัส y_i ไว้แล้ว ยังต้องจัดเก็บเวกเตอร์ระหว่างกลาง (Intermediate Vector) ทั้งหมดไว้ด้วย ดังนั้นค่าใช้จ่ายทั้งหมดในการจัดเก็บจะเพิ่มขึ้นเป็นสองเท่าดังนี้

$$M = 2N(L - 2) \dots\dots\dots (2.68)$$

ค่าใช้จ่ายที่จำเป็นข้างต้นสามารถลดลงครึ่งหนึ่งได้โดยการใช้ค่าความผิดพลาดกำลังสองเฉลี่ยในการวัดค่าความเพี้ยน ในกรณีนี้จะอาศัยการเปรียบเทียบบริเวณที่ x อยู่เมื่อเทียบกับ ระบายซึ่งแบ่งเป็นสองบริเวณ แทนที่การเปรียบเทียบ x กับเวกเตอร์ทั้งสอง โดยการเปรียบเทียบนี้จะเป็นการคำนวณผลคูณเชิงสเกลาร์ของเวกเตอร์ทั้งสองเพียงครั้งเดียวเท่านั้น

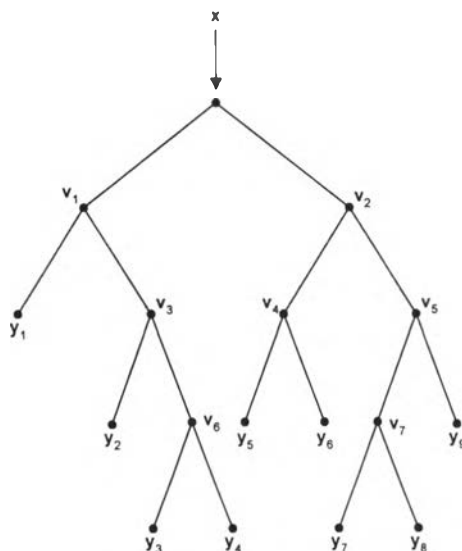


รูปที่ 2.7 แผนภูมิต้นไม้แบบสม่ำเสมอของขั้นตอนวิธีการควอนไทซ์แบบเวกเตอร์ด้วยการแบ่งแบบทวิภาค

จากรูปที่ 2.7 เป็นแผนภูมิต้นไม้เพื่อการควอนไทซ์ชนิดสม่ำเสมอ (Uniform Quantization Tree) โดยบริเวณทั้งหมดในแต่ละขั้นตอนจะถูกแบ่งออกเป็นสองบริเวณย่อย เมื่อเริ่มทำการฝึกฝนเพื่อสร้างแผนภูมิต้นไม้เพื่อการควอนไทซ์นั้น อาจแบ่งออกได้มากกว่าหนึ่งหรือสองกลุ่มย่อยที่แต่ละจุด

ในขั้นตอนการแบ่งย่อยซึ่งมีข้อมูลฝึกฝนน้อยเกินไป กลุ่มย่อยประเภทนี้จะเป็นการสูญเสียบิตข้อมูลไปโดยเปล่าประโยชน์เนื่องจากการแบ่งย่อยลงไปอีกไม่ช่วยให้ค่าความเพี้ยนลดลงไปได้อีก ดังนั้นเพื่อให้ได้ค่าความเพี้ยนเฉลี่ยที่ต่ำลงและเป็นการใช้ประโยชน์บิตข้อมูลให้ได้มากที่สุดจึงไม่ควรแบ่งแขนงของแผนภูมิต้นไม้ให้สม่ำเสมอ โดยในระหว่างการฝึกฝนที่แต่ละขั้นตอนการแบ่งย่อยนั้นจะต้องทำการตรวจสอบค่าความเพี้ยนรวมที่เกิดจากแต่ละกลุ่มเสมอ กลุ่มที่มีค่าความเพี้ยนมากที่สุดจะถูกแบ่งย่อยต่อไปและกระทำกระบวนการซ้ำอีกครั้ง ผลที่ได้จะเป็นแผนภูมิต้นไม้ชนิดไม่สม่ำเสมอ (Nonuniform Quantization Tree) ในกรณีเช่นนี้จำนวนชั้นจะเป็นจำนวนเต็มค่าใดก็ได้ โดยไม่จำกัดอยู่แต่เพียงกำลังของ 2 เท่านั้น ดังแสดงในรูปที่ 2.8 ด้วยค่า $L = 9$

การค้นหาแบบทวิภาคนี้เป็นกรณีพิเศษของวิธีการควอนไทซ์แบบเวกเตอร์ที่เรียกว่า “การควอนไทซ์แบบเวกเตอร์โดยอาศัยแผนภูมิต้นไม้ในการค้นหา” (Tree-searched Vector Quantization) ซึ่งถือว่าการค้นหาแบบทวิภาคเป็นวิธีการที่ง่ายที่สุด โดยทั่วไปแล้วจะสามารถแบ่งปริภูมิที่แต่ละขั้วในแผนภูมิต้นไม้ได้มากกว่าสองบริเวณย่อยด้วยขั้นตอนวิธีการแบ่งเฉลี่ย K ส่วนที่มี $K > 2$ วิธีการดังกล่าวนี้จะเพิ่มปริมาณการคำนวณมากกว่าการค้นหาแบบทวิภาค แต่ประสิทธิภาพของการค้นหาแบบทวิภาคก็ใกล้เคียงกับการค้นหาทั่วทั้งหมดในการประยุกต์ใช้งานทั่วไป ดังนั้นจึงควรใช้การค้นหาแบบทวิภาคแบบแผนภูมิต้นไม้ไม่สม่ำเสมอเพื่อการใช้งานที่คุ้มค่ามากที่สุด



รูปที่ 2.8 แผนภูมิต้นไม้แบบไม่สม่ำเสมอของขั้นตอนวิธีการควอนไทซ์แบบเวกเตอร์ โดยอาศัยการแบ่งแบบทวิภาค

2) ชุดรหัสสุ่ม (Random Codebooks)

นอกจากความต้องการลดค่าใช้จ่ายที่เกิดขึ้นในกระบวนการควอนไทซ์แบบเวกเตอร์แล้ว ยังต้องคำนึงถึงระยะเวลาที่ใช้ในกระบวนการฝึกฝนอีกด้วย วิธีการในการออกแบบสร้างชุดรหัสโดยไม่เสียค่าใช้จ่ายในการคำนวณระหว่างกระบวนการฝึกฝนก็คือ การสุ่มเลือกเวกเตอร์รหัสเริ่มต้นจากชุดข้อมูลฝึกฝน โดยชุดรหัสที่ถูกออกแบบสร้างด้วยวิธีการนี้เรียกว่า "ชุดรหัสสุ่ม" (Random Codebook) การนำชุดรหัสสุ่มมาใช้งานนั้นเป็นทางเลือกที่เหมาะสมในกรณีที่ค่า L และ N มีค่ามาก แต่ก็ไม่เหมาะสมในทางปฏิบัติเมื่อไม่ตรงตามเงื่อนไข อย่างไรก็ตามถึงแม้ว่าชุดรหัสสุ่มจะง่ายต่อการออกแบบสร้างแต่ก็เป็นชุดรหัสแบบค้นหาทั่วทั้งหมดซึ่งต้องใช้เวลาในการคำนวณมากและต้องการเนื้อที่เพื่อจัดเก็บชุดรหัสทั้งหมดไว้ด้วยเช่นกัน

2.3 การฝึกฝนและการทดสอบชุดรหัส (Codebook Training and Testing)

ในการออกแบบสร้างชุดรหัสได้ก็ตาม สิ่งที่สำคัญก็คือกระบวนการฝึกฝนเพื่อเสริมสร้างชุดรหัสขึ้นมา โดยขั้นตอนวิธีการแบ่งเฉลี่ย K ส่วนจะเป็นกระบวนการฝึกฝน ยกเว้นขั้นตอนกระบวนการเริ่มต้นซึ่งเป็นการเลือกชุดเวกเตอร์รหัสเริ่มต้น ในหัวข้อนี้จะกล่าวถึงกระบวนการเริ่มต้นที่ใช้ในการควอนไทซ์แบบเวกเตอร์กับสัมประสิทธิ์ของการประมาณพันธะเชิงเส้นในการเข้ารหัสเสียงพูด และจะกล่าวถึงกระบวนการทดสอบและความเสถียรของชุดรหัสที่ได้ต่อไป

2.3.1 การฝึกฝนชุดรหัส (Codebook Training)

1) การแบ่งกลุ่มแบบทวิภาค (Binary Clustering)

การค้นหาแบบทวิภาคนั้น ในแต่ละตอนจำเป็นต้องมีกระบวนการเริ่มต้นเพื่อการแบ่งปริภูมิออกเป็นสองบริเวณ โดยปกติแล้วจะอาศัยสร้างระนาบที่วิ่งผ่านค่าเฉลี่ยของชุดข้อมูลฝึกฝน และตั้งฉากกับเวกเตอร์เจาะจงที่มีค่ามากที่สุด โดยใช้ค่าความเพี้ยนกำลังสองเฉลี่ยเป็นหลัก ระนาบดังกล่าวนี้จะแบ่งปริภูมิออกเป็นสองบริเวณเริ่มต้นเพื่อเริ่มกระบวนการฝึกฝนแบบวนซ้ำ สำหรับข้อมูลที่มีมิติใดมิติหนึ่งมีค่าความแปรปรวนมากกว่ามิติอื่นจะประมาณทิศทางของเวกเตอร์เจาะจงด้วยมิติที่มีค่าความแปรปรวนมากที่สุด และนำจุดศูนย์กลางของกลุ่มทั้งสองที่ถูกแบ่งด้วยระนาบตั้งฉากมาใช้เป็นเวกเตอร์รหัสเริ่มต้นสองค่าแรก จากนั้นจึงดำเนินขั้นตอนวิธีการแบ่งเฉลี่ย K ส่วนด้วยค่า $K = 2$ เพื่อหาเวกเตอร์รหัสสุดท้าย ซึ่งโดยทั่วไปจะลู่เข้าเมื่อทำซ้ำกระบวนการไปประมาณ 5 ถึง 10 รอบ ภายหลังการแบ่งครั้งแรกสิ้นสุดลงแต่ละบริเวณย่อยจะถูกแบ่งออกเป็นสองบริเวณด้วยกระบวนการเดียวกัน

2) กระบวนการเริ่มต้นของขั้นตอนวิธีการแบ่งเฉลี่ย K ส่วน

เนื่องจากผลของขั้นตอนวิธีการแบ่งเฉลี่ย K ส่วนไม่สามารถให้ชุดรหัสที่เหมาะสมที่สุดที่ครอบคลุมทั้งหมดได้ ดังนั้นจึงต้องทำซ้ำขั้นตอนวิธีการด้วยชุดของเวกเตอร์รหัสเริ่มต้นที่แตกต่างกันไป แล้วเลือกผลของชุดรหัสที่มีค่าความเพี้ยนต่ำที่สุดมาใช้งานจริง

เนื่องจากการค้นหาแบบทวีภาคมีประสิทธิภาพใกล้เคียงกับขั้นตอนวิธีการแบ่งเฉลี่ย K ส่วนที่เหมาะสมที่สุดที่ครอบคลุมทั้งหมด ดังนั้นชุดรหัสที่ถูกสร้างขึ้นโดยการค้นหาแบบทวีภาคที่ไม่สม่ำเสมอจะเป็นชุดรหัสเริ่มต้นที่ดีสำหรับเริ่มต้นขั้นตอนวิธีการแบ่งเฉลี่ย K ส่วน การเริ่มต้นด้วยวิธีการนี้จะให้ผลลัพธ์ที่ดีกว่าการสุ่มชุดรหัสเริ่มต้น

2.3.2 การทดสอบชุดรหัส (Codebook Testing)

ภายหลังจากการออกแบบสร้างชุดรหัสด้วยชุดข้อมูลฝึกฝนแล้ว จำเป็นต้องมีการทดสอบประสิทธิภาพของชุดรหัสด้วยข้อมูลที่ไม่ขึ้นกับข้อมูลอื่นและไม่อยู่ในชุดข้อมูลฝึกฝน การทดสอบกับเพียงแค่ชุดข้อมูลฝึกฝน เป็นเพียงการแสดงวิสัยทัศน์ของชุดรหัสที่ปฏิบัติต่อข้อมูลเชิงดำเนินการเท่านั้น จาก การทดสอบชุดข้อมูลที่ไม่ขึ้นกับข้อมูลอื่นและชุดข้อมูลฝึกฝนนั้น เมื่อเพิ่มจำนวนการฝึกฝนจะมีผลให้ค่าความเพี้ยนของชุดข้อมูลทั้งสองเริ่มลู่เข้าหากันมากขึ้น แต่การลู่เข้าจะเริ่มลดลงเมื่อจำนวนของเวกเตอร์ฝึกฝนต่อชุดรหัสมีจำนวนเพิ่มขึ้นมากกว่า 20 เวกเตอร์ จึงสามารถยึดถือเป็นกฎเกณฑ์ได้ว่าการใช้ชุดข้อมูลฝึกฝน 50 ชุดต่อเวกเตอร์รหัสก็เพียงพอสำหรับการประยุกต์ใช้งานโดยทั่วไป ถ้าจำนวนข้อมูลไม่เพียงพอหรือถ้าการคำนวณหรือการจัดเก็บไม่เพียงพอ ดังนั้นการใช้ข้อมูลเพียง 10 ข้อมูลฝึกฝนต่อเวกเตอร์รหัสก็อาจจะเพียงพอแล้ว

ความแตกต่างระหว่างชุดข้อมูลที่ไม่ขึ้นกับข้อมูลอื่นและชุดข้อมูลฝึกฝนเกิดขึ้นเนื่องจากเสียงพูดที่ใช้ทดสอบแตกต่างจากเสียงพูดที่ใช้ฝึกฝน ซึ่งเกี่ยวข้องกับความทนทานของชุดรหัส โดยทั่วไปแล้วการลู่เข้าของการฝึกฝนและประสิทธิภาพที่ได้จากการทดสอบของแหล่งข้อมูลใดๆ อาจไม่เพิ่มขึ้นตามจำนวนการฝึกฝน ผลลัพธ์ที่ได้ในทางทฤษฎีเมื่อใช้การวัดค่าความเพี้ยนที่เหมาะสมจะทำให้เกิดการลู่เข้าเฉพาะกับแหล่งข้อมูลประเภทที่ไม่แปรเปลี่ยนตามเวลาโดยค่าเฉลี่ยเชิงเส้นกำกับ (Asymtotically Mean Stationary) ซึ่งไม่จำเป็นต้องเป็นไปตามเงื่อนไขการไม่แปรเปลี่ยนตามเวลา ดังนั้นเสียงพูดจากผู้พูดคนเดียวจะต้องอยู่ภายใต้ชุดเงื่อนไขสถานะแวดล้อมที่ตายตัว ตัวอย่างเช่น มีคุณสมบัติการไม่แปรเปลี่ยนตามเวลาทั้งช่วงเวลาสั้นและช่วงเวลายาว รวมทั้งถูกจำลองตามแหล่งข้อมูลที่ไม่แปรเปลี่ยนตามเวลาโดยค่าเฉลี่ยเชิงเส้นกำกับ เป็นต้น อย่างไรก็ตามแบบจำลองอาจจะไม่สามารถใช้งานได้ภายใต้เงื่อนไขที่แปรเปลี่ยนตามเวลาและมีความแปรเปลี่ยนมากขึ้น

2.3.3 ความทนทานของชุดรหัส (Codebook Robustness)

ความทนทานของชุดรหัสหมายความว่าความต้านทานของชุดรหัสที่มีต่อประสิทธิภาพที่ลดลงเมื่อทดสอบกับข้อมูลที่มีการกระจายแตกต่างไปจากข้อมูลฝึกฝน เมื่ออยู่ภายใต้เงื่อนไขเชิงดำเนินการจะไม่สามารถประมาณสถานการณ์ที่ตัวควอนไทซ์นำไปใช้งานได้ ซึ่งโดยทั่วไปแล้วการกระจายของข้อมูลเชิงดำเนินการจะแตกต่างไปจากข้อมูลฝึกฝน ดังนั้นจึงสามารถแบ่งแยกความแปรเปลี่ยนได้เป็น 2 ประเภทที่มีผลกระทบต่อารออกแบบสร้างและประสิทธิภาพเชิงดำเนินการของชุดรหัสได้แก่ (Makhoul, Roucos, and Gish, 1985) ความแปรเปลี่ยนของสัญญาณขาเข้า (Input Signal Variability) และความผิดพลาดของช่องสื่อสารเชิงเลข (Digital Transmission Channel Errors)

สำหรับเสียงพูดนั้น ความแปรเปลี่ยนของสัญญาณสามารถแบ่งย่อยออกได้เป็น ความแปรเปลี่ยนของผู้พูด (Speaker Variability) และความแปรเปลี่ยนของสภาวะแวดล้อม (Environmental Variability) ความแปรเปลี่ยนระหว่างผู้พูด (Interspeaker Variability) เกิดขึ้นเนื่องจากการเปลี่ยนแปลงเสียงพูดของผู้พูดแต่ละคน ได้แก่การเปลี่ยนแปลงปรกติธรรมดาทุกวัน การเปลี่ยนแปลงอันเนื่องมาจากสุขภาพ และการเปลี่ยนแปลงที่เกิดจากอารมณ์ เป็นต้น ความแปรเปลี่ยนภายในตัวผู้พูด (Intraspeaker Variability) หมายความว่าความแตกต่างของเสียงพูดระหว่างผู้พูดด้วยตัวเอง ความแปรเปลี่ยนของสภาวะแวดล้อม หมายความว่าระดับความดังและประเภทของเสียงรบกวนเบื้องหลังที่อยู่รอบตัวผู้พูด และลักษณะสมบัติในการรับสัญญาณซึ่งรวมไปถึงชนิดของไมโครโฟนและสิ่งอำนวยความสะดวกในการส่งผ่านสัญญาณ ประสิทธิภาพของชุดรหัสจะถูกลดทอนลงเมื่อนำมาใช้กับสัญญาณที่ไม่ได้รับการออกแบบมา ถ้าชุดรหัสที่ได้รับการออกแบบหรือฝึกฝนมาสำหรับเสียงของผู้พูดคนหนึ่ง อาจจะมีประสิทธิภาพไม่ดีสำหรับผู้พูดคนอื่นๆ ดังนั้นเพื่อให้ได้ประสิทธิภาพที่เหมาะสมที่สุดจึงควรฝึกฝนชุดรหัสโดยใช้ข้อมูลที่เทียบเท่ากับการใช้งานจริงในการฝึกฝนเท่านั้น

การลู่เข้าของค่าความเพี้ยนของชุดข้อมูลที่ไม่ขึ้นกับข้อมูลอื่นและชุดข้อมูลฝึกฝนนั้นจะลู่เข้าหากันมากขึ้นเมื่อข้อมูลที่นำมาทดสอบมาจากผู้พูดคนเดียวกันกับที่ใช้ในการฝึกฝน ค่าความเพี้ยนระหว่างข้อมูลฝึกฝนและข้อมูลทดสอบที่แตกต่างกันประมาณ 1 dB ก็ถือว่ามีความสำคัญ การเพิ่มจำนวนข้อมูลฝึกฝนจากผู้พูดเพศชาย 15 คนไม่อาจทำให้ความแตกต่างลดลงได้ เพียงแต่ทำให้ชุดรหัสที่ใช้ในการควอนไทซ์นั้นมีประสิทธิภาพดีขึ้นเมื่อใช้กับผู้พูดเพียงคนเดียวเท่านั้น ดังนั้นถ้าต้องการให้ประสิทธิภาพที่ดีที่สุดบนระบบที่ไม่ขึ้นกับผู้พูดจะต้องลดช่องว่างระหว่างชุดฝึกฝนและชุดทดสอบได้โดยการเพิ่มจำนวนผู้พูดในการฝึกฝนมากกว่าการเพิ่มจำนวนเสียงของผู้พูดแต่ละคน โดยถ้าต้องการใช้งานกับเสียงพูดผู้หญิงก็ต้องรวมเสียงพูดผู้หญิงเข้าไปในการฝึกฝนด้วยเช่นกัน

วิธีการที่เป็นไปได้ในการเพิ่มประสิทธิภาพของการควอนไทซ์แบบเวกเตอร์ให้ได้มากที่สุดนั้นก็คือ การออกแบบชุดรหัสที่ไม่ขึ้นกับผู้พูดตั้งแต่เริ่มต้นซึ่งภายหลังเมื่อใช้งานระบบจะปรับให้เข้ากับเสียงพูดของผู้พูดคนใหม่ ระบบเช่นนี้มีข้อได้เปรียบในด้านการปรับตัวโดยอัตโนมัติให้เข้ากับสภาพแวดล้อม

ของเสียงพูดของผู้พูดได้เอง การปรับตัวของชุดรหัสหมายถึงการที่เวกเตอร์รหัสเปลี่ยนไปตามเวลาซึ่งจำเป็นต้องการรับส่งเวกเตอร์รหัสชุดใหม่ไปยังตัวรับด้วยเช่นกัน ตัวอย่างเช่น ระบบแรกอาศัยชุดรหัสแบบค้นหาทั่วทั้งหมดที่ลดความเพี้ยนสูงสุดระหว่างเวกเตอร์ขาเข้าและเวกเตอร์รหัสให้มีค่าต่ำที่สุด เมื่อค่าความเพี้ยนของการควอนไทซ์สูงเกินกว่าจุดเริ่มเปลี่ยน ทำให้เวกเตอร์ขาเข้ากลายเป็นเวกเตอร์รหัสใหม่และทิ้งเวกเตอร์รหัสที่ถูกใช้งานน้อยที่สุดไป จากนั้นเวกเตอร์รหัสใหม่ที่ได้ก็จะถูกส่งไปยังตัวรับ เป็นต้น

ค. การทดสอบความคล้ายคลึงกันของรูปแบบ (Pattern Similarity Testing)

การทดสอบความคล้ายคลึงกันของรูปแบบ (Pattern Similarity Testing) หรือการจำแนกรูปแบบ (Pattern Classification) เกี่ยวข้องกับการเปรียบเทียบรูปแบบเสียงพูดระหว่างคำพูดหรือวลีที่ไม่ทราบรูปแบบกับรูปแบบที่ได้จัดเก็บไว้แล้ว ซึ่งก็คือชุดรูปร่างต้นแบบ (Templates) หรือแบบจำลอง (Models) ของเสียงพูด ภายหลังการเปรียบเทียบแต่ละครั้งจะได้ค่าความไม่คล้ายคลึงกัน (Dissimilarity Scores) หรือค่าระยะทาง (Distance Scores) เพื่อใช้ในขั้นตอนวิธีการตัดสินใจในการเลือกรูปแบบที่เหมาะสมใกล้เคียงที่สุดต่อไป

ขั้นตอนวิธีการในการจำแนกรูปแบบที่ถูกนำมาใช้มากที่สุด สามารถแบ่งออกได้เป็น 4 วิธีการ (Roe and Wilpon, 1993) ได้แก่ การเข้าคู่ต้นแบบ (Template Matching) ระบบตามกฎเกณฑ์ (Rule-Based System) ระบบแบบจำลองฮิดเดน มาร์คอฟ (Hidden Markov Model, HMM) และเครือข่ายประสาท (Neural Network) แต่เนื่องจากงานวิจัยนี้ใช้เทคนิคระบบแบบจำลองฮิดเดน มาร์คอฟ จึงจะกล่าวถึงแต่เฉพาะรายละเอียดและทฤษฎีของวิธีการจำแนกรูปแบบวิธีนี้เท่านั้น

ง. ระบบแบบจำลองฮิดเดน มาร์คอฟ (Hidden Markov Model)

ระบบแบบจำลองฮิดเดน มาร์คอฟนี้ถือเป็นขั้นตอนวิธีการจำแนกรูปแบบที่ดีที่สุดวิธีการหนึ่งที่มีอยู่ในขณะนี้โดยอาศัยวิธีการทางสถิติ (Roe and Wilpon, 1993) ขั้นตอนวิธีการนี้มีข้อได้เปรียบที่สำคัญเหนือวิธีการเข้าคู่ต้นแบบก็คือ สามารถเก็บข้อมูลรายละเอียดในทางสถิติเกี่ยวกับเสียงพูดไว้ได้มากกว่าวิธีการเข้าคู่ต้นแบบ โดยเก็บข้อมูลการกระจายที่สมบูรณ์ของลักษณะสำคัญที่มีอยู่ในข้อมูลฝึกฝน จึงสามารถจำแนกความแตกต่างระหว่างเสียงพูดได้ดีมากยิ่งขึ้น อีกทั้งขั้นตอนวิธีการนี้ยังอาศัยการโปรแกรมแบบพลวัต (Dynamic Programming) ทำให้มีความรวดเร็วในการประมวลผลมากยิ่งขึ้น

เหตุผลที่ระบบแบบจำลองฮิดเดน มาร์คอฟเป็นที่นิยมมีด้วยกัน 2 ประการ (Rabiner, 1989) ประการแรก แบบจำลองนี้อาศัยโครงสร้างทางคณิตศาสตร์และสามารถเปลี่ยนแปลงทฤษฎีพื้นฐานเพื่อประยุกต์ใช้งานได้อย่างกว้างขวาง ประการที่สอง แบบจำลองนี้สามารถทำงานได้เป็นอย่างดีเมื่อประยุกต์ใช้งานอย่างเหมาะสม

สำหรับกระบวนการโดยทั่วไปจะสร้างผลลัพธ์ที่สังเกตได้ซึ่งสามารถแสดงได้เป็นสัญญาณ สัญญาณอาจมีความไม่ต่อเนื่องโดยธรรมชาติหรือมีความต่อเนื่องโดยธรรมชาติก็ได้ แหล่งของสัญญาณอาจเป็นได้ทั้งที่ไม่แปรเปลี่ยนตามเวลาหรือแปรเปลี่ยนตามเวลาก็ได้ ปัญหาที่สำคัญก็คือการสร้างแบบจำลองสัญญาณขึ้นมา โดยมีเหตุผล 3 ประการ (Rabiner, 1989) ดังนี้ ประการแรก แบบจำลองของสัญญาณเป็นพื้นฐานของรายละเอียดในทางทฤษฎีของระบบประมวลผลสัญญาณ ซึ่งใช้ในการประมวลผลสัญญาณเพื่อให้ได้ผลลัพธ์ตามต้องการ ประการที่สอง ช่วยในการศึกษาแหล่งของสัญญาณโดยปราศจากแหล่งของสัญญาณ ประการที่สาม สามารถใช้งานได้เป็นอย่างดีในทางปฏิบัติ

ประเภทของแบบจำลองสัญญาณสามารถแบ่งได้เป็น 2 ประเภท ได้แก่ แบบจำลองที่กำหนดการ (Deterministic Models) และแบบจำลองทางสถิติ (Statistical Models) แบบจำลองที่กำหนดการจะบอกถึงคุณสมบัติเฉพาะของสัญญาณ โดยอาศัยเพียงการประมาณค่าพารามิเตอร์ที่จำเป็นให้แก่แบบจำลองสัญญาณเท่านั้น ส่วนแบบจำลองทางสถิติจะอาศัยคุณสมบัติทางสถิติของสัญญาณในการบอกคุณสมบัติของสัญญาณ โดยอาศัยสมมติฐานที่ว่าสัญญาณสามารถแสดงได้ด้วยกระบวนการสุ่มแบบพาราเมตริก และค่าพารามิเตอร์ของกระบวนการสุ่มสามารถประมาณค่าได้อย่างแม่นยำ

ส่วนแบบจำลองฮิดเดน มาร์คอฟจัดอยู่ในประเภทหนึ่งของแบบจำลองสัญญาณสุ่ม โดยการออกแบบเป็นการแก้ไขปัญหาคำถามพื้นฐานสำคัญ 3 ประการของแบบจำลองฮิดเดน มาร์คอฟ (Rabiner, 1989) ได้แก่ การประเมินค่าความน่าจะเป็นของลำดับค่าสังเกตสำหรับเฉพาะแบบจำลองฮิดเดน มาร์คอฟ การหาลำดับที่ดีที่สุดสำหรับแต่ละสถานะของแบบจำลอง และการปรับค่าพารามิเตอร์เพื่อให้เหมาะสมที่สุดกับสัญญาณที่สังเกต

1. องค์ประกอบของแบบจำลองฮิดเดน มาร์คอฟ

องค์ประกอบของแบบจำลองฮิดเดน มาร์คอฟ ประกอบไปด้วยพารามิเตอร์ต่างๆ ดังนี้

1) N คือจำนวนสถานะที่อยู่ภายในแบบจำลอง ซึ่งโดยทั่วไปแล้วแต่ละสถานะจะเชื่อมโยงถึงกันด้วยวิธีการที่จะทำให้สถานะใดๆ สามารถเข้าถึงสถานะอื่นๆ ได้ แต่ละสถานะแสดงได้ด้วย $S = \{S_1, S_2, \dots, S_N\}$ โดยมีสถานะที่เวลา t แสดงได้ด้วย q_t

2) M คือจำนวนสัญลักษณ์ของค่าสังเกตต่อสถานะ ซึ่งสัญลักษณ์ของค่าสังเกตจะสัมพันธ์กับผลลัพธ์ ขาออกทางกายภาพของระบบที่ถูกจำลอง แต่ละสัญลักษณ์สามารถแสดงได้ด้วย $V = \{v_1, v_2, \dots, v_M\}$

3) การกระจายของความน่าจะเป็นในการเปลี่ยนแปลงสถานะ (State Transition Probability Distribution) $A = \{a_{ij}\}$ เมื่อ

$$a_{ij} = P[q_{t+1} = S_j | q_t = S_i], \quad 1 \leq i, j \leq N \dots\dots\dots (2.69)$$

ในกรณีเฉพาะที่สถานะใดๆ สามารถเข้าถึงสถานะอื่นได้ภายในขั้นตอนเดียวจะกำหนดให้ $a_{ij} > 0, \forall i, j$ ส่วนในกรณีอื่นนอกเหนือจากนี้จะกำหนดให้ $a_{ij} = 0$ สำหรับ (i, j) เพียงคู่เดียวหรือมากกว่า

4) การกระจายของความน่าจะเป็นของสัญลักษณ์ของค่าสังเกต (Observation Symbol Probability Distribution) $B = \{b_j(k)\}$ ในสถานะที่ j เมื่อ

$$b_j(k) = P[v_k \text{ at } t | q_t = S_j], \quad \begin{matrix} 1 \leq j \leq N \\ 1 \leq k \leq M \end{matrix} \quad (2.70)$$

5) การกระจายของสภาวะเริ่มต้น (Initial State Distribution) $\pi = \{\pi_i\}$ เมื่อ

$$\pi_i = P[q_1 = S_i], \quad 1 \leq i \leq N \quad (2.71)$$

โดยการกำหนดค่าที่เหมาะสมให้กับองค์ประกอบ N, M, A, B, π ของแบบจำลองฮิดเดน มาร์คอฟ ซึ่งใช้ในการกำเนิดลำดับค่าสังเกต เมื่อแต่ละค่าสังเกต O_t เป็นสัญลักษณ์ที่ได้จาก V และ T เป็นจำนวนค่าสังเกตทั้งหมดที่มีในลำดับ ซึ่งมีขั้นตอนวิธีการดังนี้

$$O = O_1 O_2 \cdots O_T \quad (2.72)$$

ตารางที่ 2.3 รายละเอียดของขั้นตอนในการกำเนิดลำดับค่าสังเกต

ขั้นตอนที่ 1	เลือกสถานะเริ่มต้น $q_1 = S_i$ ที่สัมพันธ์กับการกระจายของสภาวะเริ่มต้น π
ขั้นตอนที่ 2	กำหนดให้ $t = 1$
ขั้นตอนที่ 3	เลือก $O_t = v_k$ ที่สัมพันธ์กับการกระจายของความน่าจะเป็นของสัญลักษณ์เมื่ออยู่ในสถานะ S_i เช่น $b_i(k)$
ขั้นตอนที่ 4	เคลื่อนย้ายไปยังสถานะใหม่ $q_{t+1} = S_j$ ที่สัมพันธ์กับการกระจายของความน่าจะเป็นในการเปลี่ยนแปลงสถานะสำหรับสถานะ S_i เช่น a_{ij}
ขั้นตอนที่ 5	กำหนดให้ $t = t + 1$ แล้วกลับไปทำซ้ำขั้นตอนที่ 3 ใหม่ถ้า $t < T$ นอกเหนือจากนี้ให้ยุติกระบวนการ

ขั้นตอนดังกล่าวนี้สามารถได้ทั้งการกำเนิดค่าสังเกต และเป็นแบบจำลองเพื่อบอกถึงความเหมาะสมในการกำเนิดลำดับค่าสังเกตด้วยแบบจำลองฮิดเดน มาร์คอฟ ดังนั้นการกำหนดคุณสมบัติเฉพาะของแบบจำลองฮิดเดน มาร์คอฟต้องการคุณสมบัติเฉพาะของพารามิเตอร์ของแบบจำลองสองค่า (N และ M)

คุณสมบัติเฉพาะของสัญลักษณ์ของค่าสังเกต และคุณสมบัติเฉพาะของการวัดค่าความน่าจะเป็นได้แก่ A, B, π โดยเขียนอยู่ในรูปแบบย่อเพื่อบ่งบอกชุดของพารามิเตอร์ที่สมบูรณ์ของแบบจำลองดังนี้

$$\lambda = (A, B, \pi) \dots\dots\dots (2.73)$$

2. ปัญหาพื้นฐานสามประการของแบบจำลองฮิดเดน มาร์คอฟ

ในการประยุกต์ใช้งานแบบจำลองฮิดเดน มาร์คอฟในทางปฏิบัตินั้น ก็คือการแก้ไขปัญหาคือพื้นฐานทั้งสามประการ โดยมีรายละเอียดดังแสดงในตารางที่ 2.2

ปัญหาพื้นฐานข้อแรก คือปัญหาในการประเมินค่า ซึ่งก็คือการหาค่าความน่าจะเป็นของลำดับค่าสังเกตที่สร้างจากแบบจำลอง เมื่อกำหนดแบบจำลองและลำดับค่าสังเกตมาให้ หรืออีกนัยหนึ่งก็คือการแสดงให้เห็นว่าแบบจำลองที่กำหนดให้เข้าคู่กันได้ดีกับลำดับค่าสังเกตที่กำหนดให้ได้ดีเพียงใด ตัวอย่างเช่น ถ้าในกรณีที่เราพิจารณาเลือกระหว่างแบบจำลองหลายแบบ ผลลัพธ์ของปัญหาพื้นฐานข้อแรกจะช่วยให้การเลือกแบบจำลองที่เข้าคู่กันได้ดีที่สุดกับค่าสังเกต

ปัญหาพื้นฐานข้อที่สอง คือความพยายามในการเปิดเผยส่วนที่แบบจำลองปิดบังไว้ ในเฉพาะกรณีของแบบจำลองที่ด้อยประสิทธิภาพจะไม่มีลำดับสถานะที่ถูกต้อง ดังนั้นในทางปฏิบัติจึงใช้กฎเกณฑ์ของความเหมาะสมที่สุดในการแก้ไขปัญหานี้ซึ่งมีด้วยกันหลายประเภท ดังนั้นการเลือกกฎเกณฑ์จึงเท่ากับเป็นการเปิดเผยลำดับสถานะที่ถูกปกปิดโดยแบบจำลอง

ปัญหาพื้นฐานข้อที่สาม คือการทำให้พารามิเตอร์ของแบบจำลองมีประสิทธิภาพมากที่สุด เพื่อที่จะอธิบายลำดับค่าสังเกตได้ดีที่สุด ลำดับค่าสังเกตที่ใช้ในการปรับพารามิเตอร์ของแบบจำลองเรียกว่า "ลำดับฝึกฝน" (Training Sequences) เนื่องจากถูกใช้ในการฝึกฝนแบบจำลองฮิดเดน มาร์คอฟ ปัญหาในการฝึกฝนนี้จะช่วยให้ปรับแต่งพารามิเตอร์ของแบบจำลองให้เหมาะสมมากที่สุดกับข้อมูลฝึกฝนที่สังเกต

ในการประยุกต์ใช้งานแบบจำลองฮิดเดน มาร์คอฟกับการรู้จำคำพูดนั้น เริ่มต้นจากการออกแบบสร้างแบบจำลองฮิดเดน มาร์คอฟ N สถานะสำหรับแต่ละคำของชุดคำศัพท์ W คำ สัญลักษณ์เสียงพูดของแต่ละคำจะถูกแทนที่ด้วยลำดับเวลาของเวกเตอร์ที่สเปกตรัม การเข้ารหัสจะอาศัยชุดรหัสสเปกตรัมที่ประกอบด้วยเวกเตอร์เชิงสเปกตรัม M เวกเตอร์ที่เป็นเอกลักษณ์ จึงทำให้แต่ละค่าสังเกตจะเป็นดรรชนีของเวกเตอร์เชิงสเปกตรัมที่ใกล้เคียงกับสัญญาณเสียงพูดต้นฉบับมากที่สุด ดังนั้นแต่ละคำศัพท์จะมีลำดับการฝึกฝนที่ประกอบด้วยจำนวนลำดับของดรรชนีชุดรหัสของคำ ขั้นตอนแรกเริ่มจากการสร้างแบบจำลองของแต่ละคำโดยการแก้ปัญหาคือพื้นฐานข้อที่ 3 เพื่อประมาณค่าพารามิเตอร์ของแบบจำลองให้เหมาะสมที่สุดสำหรับแต่ละแบบจำลอง ขั้นตอนที่สองเป็นการสร้างความเข้าใจในความหมายทางกายภาพของสถานะของแบบจำลองโดยการแก้ปัญหาคือพื้นฐานข้อที่ 2 เพื่อแบ่งแยกแต่ละลำดับฝึกฝนของคำไปยังแต่ละสถานะและศึกษาถึงคุณสมบัติของเวกเตอร์เชิงสเปกตรัมที่ทำให้เกิดค่าสังเกตในแต่ละสถานะ โดยในขั้นตอนนี้จะทำการปรับแต่งแบบจำลองเพื่อเพิ่มพูนความสามารถในการจำลองแบบลำดับคำพูด ขั้นตอนที่สามทำภายหลังจาก

การออกแบบชุดของแบบจำลองฮิดเดน มาร์คอฟทั้ง W ชุดพร้อมทั้งปรับให้มีประสิทธิภาพที่เหมาะสมแล้ว การรู้จำคำพูดที่ไม่รู้จักมาก่อนจะอาศัยการแก้ปัญหาพื้นฐานข้อที่ 1 เพื่อให้คะแนนแต่ละแบบจำลองของคำพูด ด้วยลำดับค่าสังเกตที่ใช้ทดสอบและเลือกค่าซึ่งมีแบบจำลองที่ให้คะแนนสูงที่สุด

ตารางที่ 2.4 รายละเอียดของปัญหาพื้นฐานสามประการของแบบจำลองฮิดเดน มาร์คอฟ

ปัญหาพื้นฐานข้อที่ 1	เมื่อกำหนดลำดับค่าสังเกต $O = O_1 O_2 \dots O_T$ และแบบจำลอง $\lambda = (A, B, \pi)$ จะทำการหาค่าความน่าจะเป็นของลำดับค่าสังเกต $P(O \lambda)$ ตามแบบจำลองที่กำหนดให้ได้อย่างไร
ปัญหาพื้นฐานข้อที่ 2	เมื่อกำหนดลำดับค่าสังเกต $O = O_1 O_2 \dots O_T$ และแบบจำลอง λ จะทำการเลือกลำดับสถานะที่สัมพันธ์กับ $Q = q_1 q_2 \dots q_T$ ซึ่งมีความเหมาะสมที่สุดกับแบบจำลองที่กำหนดให้ได้อย่างไร
ปัญหาพื้นฐานข้อที่ 3	จะทำการปรับค่าพารามิเตอร์ของแบบจำลอง $\lambda = (A, B, \pi)$ อย่างไร เพื่อให้ค่าความน่าจะเป็นของลำดับค่าสังเกต $P(O \lambda)$ มีค่ามากที่สุด

3. การแก้ไขปัญหาพื้นฐานสามประการของแบบจำลองฮิดเดน มาร์คอฟ

ในการแก้ไขปัญหามูลฐานทั้งสามประการของแบบจำลองฮิดเดน มาร์คอฟ เพื่อใช้ในการรู้จำคำพูดตามแนวทางดังกล่าวข้างต้นสามารถแสดงได้ดังนี้

3.1 การแก้ไขปัญหาพื้นฐานข้อที่ 1

เมื่อกำหนดแบบจำลอง λ การหาค่าความน่าจะเป็น $P(O|\lambda)$ ของลำดับค่าสังเกต $O = O_1 O_2 \dots O_T$ จะอาศัยวิธีการหาค่าทุกลำดับสถานะความยาว T ที่เป็นไปได้ทั้งหมดตามลำดับ เมื่อ T เป็นจำนวนค่าสังเกต พิจารณาลำดับของสถานะที่มีค่าจำกัดดังนี้

$$Q = q_1 q_2 \dots q_T \dots\dots\dots (2.74)$$

เมื่อ q_1 เป็นสถานะเริ่มต้น ค่าความน่าจะเป็นของลำดับค่าสังเกต O สำหรับลำดับสถานะในสมการที่ (2.69) ดังนี้

$$P(O|Q, \lambda) = \prod_{i=1}^T P(O_i|q_i, \lambda) \dots\dots\dots (2.75)$$

เมื่อกำหนดให้ค่าสังเกตเป็นชนิดไม่ขึ้นแก่กันในทางสถิติจะได้ว่า

$$P(O|Q, \lambda) = b_{q_1}(O_1) \cdot b_{q_2}(O_2) \dots b_{q_T}(O_T) \dots\dots\dots (2.76)$$

ดังนั้นค่าความน่าจะเป็นของลำดับสถานะ Q สามารถเขียนได้เป็น

$$P(Q|\lambda) = \pi_{q_1} a_{q_1 q_2} a_{q_2 q_3} \cdots a_{q_{T-1} q_T} \dots (2.77)$$

ค่าความน่าจะเป็นร่วมระหว่าง O และ Q เป็นเพียงผลคูณของสมการข้างต้นดังนี้

$$P(O, Q|\lambda) = P(O|Q, \lambda) P(Q, \lambda) \dots (2.78)$$

ค่าความน่าจะเป็นของ O เมื่อกำหนดแบบจำลองให้ สามารถหาได้โดยผลรวมของค่าความน่าจะเป็นร่วมของลำดับสถานะทั้งหมดที่เป็นไปได้ q ดังนี้

$$\begin{aligned} P(Q|\lambda) &= \sum_{\text{all } Q} P(O|Q, \lambda) P(Q, \lambda) \dots (2.79) \\ &= \sum_{q_1, q_2, \dots, q_T} \pi_{q_1} b_{q_1}(O_1) a_{q_1 q_2} b_{q_2}(O_2) \cdots a_{q_{T-1} q_T} b_{q_T}(O_T) \end{aligned}$$

จากสมการที่ (2.79) เริ่มต้นที่เวลา $t = 1$ จะเริ่มที่สถานะ q_1 ด้วยค่าความน่าจะเป็น π_{q_1} และให้กำเนิดสัญลักษณ์ O_1 ภายในสถานะเดียวกันด้วยค่าความน่าจะเป็น $b_{q_1}(O_1)$ เมื่อเวลาเปลี่ยนจาก t เป็น $t + 1$ ($t = 2$) และเปลี่ยนไปยังสถานะ q_2 จากสถานะ q_1 ด้วยค่าความน่าจะเป็น $a_{q_1 q_2}$ และให้กำเนิดสัญลักษณ์ O_2 ภายในสถานะเดียวกันด้วยค่าความน่าจะเป็น $b_{q_2}(O_2)$ กระบวนการนี้จะดำเนินไปอย่างต่อเนื่องจนกระทั่งถึงการเปลี่ยนแปลงที่เวลา T จากสถานะ q_{T-1} ไปยังสถานะ q_T ด้วยค่าความน่าจะเป็น $a_{q_{T-1} q_T}$ และให้กำเนิดสัญลักษณ์ O_T ภายในสถานะเดียวกันด้วยค่าความน่าจะเป็น $b_{q_T}(O_T)$

ในการคำนวณค่าความน่าจะเป็น $P(O|\lambda)$ ตามสมการที่ (2.79) นั้นจะเกิดการคำนวณขึ้นด้วยอันดับประมาณ $2T \cdot N^T$ ครั้ง เนื่องจากทุกเวลา $t = 1, 2, \dots, T$ จะเกิดสถานะที่เป็นไปได้ N สถานะซึ่งเข้าถึงได้ และในแต่ละลำดับสถานะจะเกิดการคำนวณขึ้นประมาณ $2T$ ครั้งสำหรับแต่ละพจน์ในผลรวมของสมการที่ (2.79) ซึ่งจำแนกได้เป็นการคูณ $(2T - 1)N^T$ ครั้งและการบวก $N^T - 1$ ครั้ง จึงทำให้การคำนวณค่าโดยใช้สมการนี้เป็นไปไม่ได้ถึงแม้ด้วย N และ T ค่าน้อยๆ ก็ตาม ดังนั้นจึงมีกระบวนการที่มีประสิทธิภาพในการคำนวณค่าที่เรียกว่า "กระบวนการไปหน้า-ย้อนกลับ" (Forward-Backward Procedure) ซึ่งประกอบด้วยกระบวนการไปหน้าและกระบวนการย้อนกลับร่วมกันดังนี้

1) กระบวนการไปหน้า (Forward Procedure)

พิจารณาตัวแปรไปหน้า $\alpha_t(i)$ ที่กำหนดไว้แล้วดังนี้

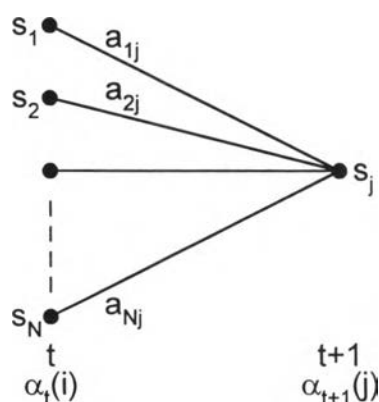
$$\alpha_t(i) = P(O_1 O_2 \cdots O_t, q_t = S_i | \lambda) \dots (2.80)$$

ค่าความน่าจะเป็นของลำดับค่าสังเกตบางส่วน $O_1 O_2 \cdots O_t$ และสถานะ S_i ที่เวลา t เมื่อกำหนดแบบจำลอง λ จะสามารถหาค่าของ $\alpha_t(i)$ โดยอุปนัยได้ในตารางที่ 2.5 ซึ่งแสดงถึงรายละเอียดของกระบวนการไปหน้าดังนี้

ตารางที่ 2.5 รายละเอียดกระบวนการไปหน้า

ขั้นตอนที่ 1 กระบวนการเริ่มต้น	$\alpha_1(i) = \pi_i b_i(O_1),$ $1 \leq i \leq N$	(2.81)
ขั้นตอนที่ 2 กระบวนการอุปนัย	$\alpha_{t+1}(j) = \left[\sum_{i=1}^N \alpha_t(i) a_{ij} \right] b_j(O_{t+1}),$ $1 \leq t \leq T-1, 1 \leq i \leq N$	(2.82)
ขั้นตอนที่ 3 กระบวนการสิ้นสุด	$P(O \lambda) = \sum_{i=1}^N \alpha_T(i)$	(2.83)

ขั้นตอนกระบวนการเริ่มต้น เป็นการกำหนดค่าเริ่มต้นให้กับค่าความน่าจะเป็นแบบไปหน้า โดยกำหนดให้เป็นความน่าจะเป็นร่วมของสถานะ S_i และค่าสังเกตเริ่มต้น O_1 ขั้นตอนกระบวนการอุปนัยถือเป็นหัวใจสำคัญของกระบวนการไปหน้าดังแสดงในรูปที่ 2.9 ซึ่งแสดงถึงการที่สถานะ S_j ที่สามารถเข้าถึงได้ทีเวลา $t + 1$ จาก N สถานะที่เป็นไปได้ $S_i, 1 \leq i \leq N$ ที่เวลา t เนื่องจาก $\alpha_t(i)$ เป็นค่าความน่าจะเป็นของเหตุการณ์ร่วมที่สังเกต $O_1 O_2 \dots O_t$ และสถานะที่เวลา t เป็น S_i ดังนั้นผลคูณ $\alpha_t(i) a_{ij}$ จึงเป็นค่าความน่าจะเป็นของเหตุการณ์ร่วมที่สังเกต $O_1 O_2 \dots O_t$ และเข้าถึงสถานะ S_j ที่เวลา $t + 1$ ผ่านทางสถานะ S_i ที่เวลา t ผลรวมผลคูณของ N สถานะทั้งหมดที่เป็นไปได้ $S_i, 1 \leq i \leq N$ ที่เวลา t ได้ผลเป็นค่าความน่าจะเป็น S_j ที่เวลา $t + 1$ ร่วมกับค่าสังเกตบางส่วนทั้งหมดที่สัมพันธ์กัน

รูปที่ 2.9 รายละเอียดลำดับกระบวนการในการคำนวณค่าตัวแปรไปหน้า $\alpha_t(i)$

เมื่อเสร็จสิ้นกระบวนการทั้งหมดและได้ค่า S_j โดยค่า $\alpha_{t+1}(j)$ จะได้มาจากค่าสังเกต O_{t+1} ในสถานะ j การคำนวณในสมการที่ (2.82) จะกระทำกับทุกสถานะ $j, 1 \leq j \leq N$ เมื่อกำหนดค่า t ให้ จากนั้นจะวนซ้ำกับทุก $t = 1, 2, \dots, T-1$ ขั้นตอนกระบวนการสิ้นสุด เป็นการคำนวณค่า $P(O|\lambda)$ ด้วยผลรวมของตัวแปรไปหน้าตัวสุดท้าย $\alpha_T(i)$ ซึ่งทำให้ $P(O|\lambda)$ เป็นเพียงผลรวมของ $\alpha_T(i)$ แต่ละตัวมีนิยามดังนี้

$$\alpha_T(i) = P(O_1 O_2 \dots O_T, q_T = S_i | \lambda) \dots\dots\dots (2.84)$$

เมื่อพิจารณาการคำนวณค่า $\alpha_t(j), 1 \leq t \leq T, 1 \leq j \leq N$ นั้น จะเกิดการคำนวณขึ้นด้วยอันดับประมาณ $N^2 T$ ครั้งเมื่อเปรียบเทียบกับ $2TN^T$ ครั้งเมื่อคำนวณโดยตรง ซึ่งจำแนกได้เป็นการคูณ $N(N+1)(T-1) + N$ ครั้งและการบวก $N(N-1)(T-1)$ ครั้ง

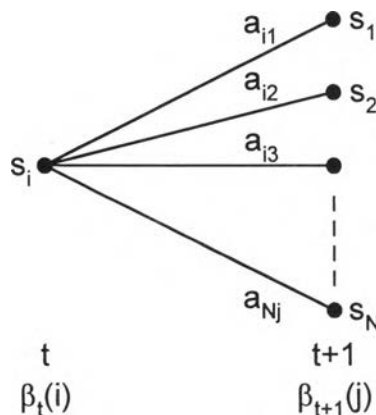
การคำนวณค่าความน่าจะเป็นแบบไปหน้าจะอยู่บนพื้นฐานของโครงสร้าง Lattice หรือ Trellis กฎแจลาคัญก็คือเนื่องจากมีเพียง N สถานะดังนั้นลำดับสถานะทั้งหมดที่เป็นไปได้จะรวมเข้ากับ N ปมในโครงสร้างไม่ว่าลำดับของค่าสังเกตจะยาวเพียงใดก็ตาม ที่เวลา $t=1$ ซึ่งเป็นช่วงเวลาแรกของโครงสร้าง Lattice จะทำการคำนวณค่าของ $\alpha_1(i), 1 \leq i \leq N$ ที่เวลา $t=2, 3, \dots, T$ จะเป็นเพียงการคำนวณค่าของ $\alpha_t(j), 1 \leq j \leq N$ โดยแต่ละครั้งจะคำนวณเพียงค่า $\alpha_{t-1}(i)$ จำนวน N ค่าก่อนหน้านั้นเท่านั้น เนื่องจากแต่ละจุดตาราง N จุดสามารถเข้าถึงได้จาก N จุดตารางเดียวกันของช่วงเวลาก่อนหน้านั้น

2) กระบวนการย้อนกลับ (Backward Procedure)

พิจารณาตัวแปรย้อนกลับ $\beta_t(i)$ ที่กำหนดไว้แล้วดังนี้

$$\beta_t(i) = P(O_{t+1} O_{t+2} \dots O_T | q_t = S_i, \lambda) \dots\dots\dots (2.85)$$

ค่าความน่าจะเป็นของลำดับค่าสังเกตบางส่วนจาก $t+1$ จนถึงสิ้นสุดเมื่อกำหนดสถานะ S_i ที่เวลา t ด้วยแบบจำลอง λ จะสามารถหาค่าของ $\beta_t(i)$ โดยอุปนัยได้ดังนี้



รูปที่ 2.10 รายละเอียดลำดับกระบวนการในการคำนวณค่าตัวแปรย้อนกลับ $\beta_t(i)$

ตารางที่ 2.6 รายละเอียดกระบวนการย้อนกลับ

ขั้นตอนที่ 1 กระบวนการเริ่มต้น	$\beta_T(i) = 1, \quad 1 \leq i \leq N$ (2.86)
ขั้นตอนที่ 2 กระบวนการอุปนัย	$\beta_t(i) = \sum_{j=1}^N a_{ij} b_j(O_{t+1}) \beta_{t+1}(j),$ (2.87) $t = T-1, T-2, \dots, 1, \quad 1 \leq i \leq N$
ขั้นตอนที่ 3 กระบวนการสิ้นสุด	$P(O \lambda) = \sum_{i=1}^N \pi_i b_i(O_1) \beta_1(i)$ (2.88)

ขั้นตอนกระบวนการเริ่มต้น เป็นการกำหนดค่าเริ่มต้นให้กับ $\beta_T(i)$ โดยกำหนดให้เป็น 1 ทั้งหมดทุกค่า i ขั้นตอนกระบวนการอุปนัยดังแสดงในรูปที่ 2.10 แสดงถึงการเข้าถึงสถานะ S_j ที่เวลา t และการให้รายละเอียดของลำดับค่าสังเกตตั้งแต่วเวลา $t+1$ เป็นต้นไป จะต้องพิจารณาสถานะ S_j ที่เป็นไปได้ทั้งหมด เพื่อนับรวมการเปลี่ยนแปลงจากสถานะ S_i ไปยัง S_j ของพจน์ a_{ij} รวมไปถึงค่าสังเกต O_{t+1} ในสถานะ j ของพจน์ $b_j(O_{t+1})$ และนับรวมถึงลำดับค่าสังเกตบางส่วนที่ยังเหลืออยู่จากสถานะ j ของพจน์ $\beta_{t+1}(j)$ ด้วย ซึ่งในการคำนวณค่า $\beta_t(i), 1 \leq t \leq T, 1 \leq i \leq N$ นั้น จะเกิดการคำนวณขึ้นด้วยอันดับประมาณ $N^2 T$ ครั้ง และสามารถคำนวณโดยอาศัยโครงสร้าง Lattice ได้เช่นเดียวกัน

3.2 การแก้ไขปัญหาค่าพื้นฐานข้อที่ 2

ปัญหาค่าพื้นฐานข้อที่ 2 เกี่ยวข้องกับการหาลำดับสถานะที่เหมาะสมที่สุดที่สัมพันธ์กับลำดับค่าสังเกตที่กำหนดให้ ความยากในการแก้ปัญหาค่าขึ้นอยู่กับนิยามของลำดับสถานะที่เหมาะสมที่สุด ซึ่งมีกฎเกณฑ์ของความเหมาะสมที่สุดที่เป็นไปได้มากมาย กฎเกณฑ์หนึ่งที่เป็นไปได้ก็คือ การเลือกสถานะ q_t ซึ่งตัวต่อตัวแล้วคล้ายคลึงกันมากที่สุด กฎเกณฑ์ของความเหมาะสมที่สุดนี้จะทำให้ค่าประมาณของจำนวนสถานะที่ถูกต้องมีค่ามากที่สุด กำหนดให้ตัวแปรที่มีค่าดังนี้

$$\gamma_t(i) = P(q_t = S_i | O, \lambda) \dots\dots\dots (2.89)$$

ซึ่งเป็นค่าความน่าจะเป็นของการอยู่ในสถานะ S_i ที่เวลา t เมื่อกำหนดลำดับค่าสังเกต O และแบบจำลอง λ มาให้ สมการที่ (2.84) สามารถเขียนอยู่ในรูปของตัวแปรไปหน้าและตัวแปรย้อนกลับได้ดังนี้

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{P(O|\lambda)} = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^N \alpha_t(i)\beta_t(i)} \dots\dots\dots (2.90)$$

เมื่อ $\alpha_i(i)$ เป็นลำดับค่าสังเกตบางส่วน $O_1 O_2 \dots O_t$ และสถานะ S_t ที่เวลา t ขณะที่ $\beta_i(i)$ เป็นลำดับค่าสังเกตที่คงเหลืออยู่ $O_{t+1} O_{t+2} \dots O_T$ เมื่อกำหนดสถานะ S_t ที่เวลา t ตัวประกอบที่ทำให้เป็นบรรทัดฐานเดียวกัน $P(O|\lambda) = \sum_{i=1}^N \alpha_i(i)\beta_i(i)$ ทำให้ $\gamma_i(i)$ กลายเป็นการวัดค่าความน่าจะเป็นดังนี้

$$\sum_{i=1}^N \gamma_i(i) = 1, \quad 1 \leq t \leq T \dots\dots\dots (2.91)$$

โดยการใช้ $\gamma_i(i)$ จะสามารถหาค่าสถานะที่ตัวต่อตัวคล้ายคลึงกันมากที่สุด q_t ที่เวลา t ได้ดังนี้

$$q_t = \arg \max_{1 \leq i \leq N} [\gamma_i(i)], \quad 1 \leq t \leq T \dots\dots\dots (2.92)$$

แม้ว่าสมการที่ (2.92) จะทำให้ค่าประมาณของจำนวนสถานะที่ถูกต้องมีค่ามากที่สุดโดยการเลือกสถานะที่คล้ายคลึงกันมากที่สุดสำหรับแต่ละ t ก็ตาม ก็ยังเกิดปัญหาขึ้นกับผลของลำดับสถานะที่ได้ เนื่องจากสมการที่ (2.92) เป็นเพียงการหาสถานะที่คล้ายคลึงกันมากที่สุดในขณะหนึ่งเท่านั้น โดยไม่เกี่ยวข้องกับค่าความน่าจะเป็นในการปรากฏของลำดับสถานะ

วิธีการหนึ่งในการแก้ปัญหา ก็คือการแก้ไขกฎเกณฑ์ของความเหมาะสมที่สุด โดยการคำนวณหาลำดับสถานะที่ทำให้ค่าประมาณของคู่สถานะ (q_t, q_{t+1}) ที่ถูกต้องมีค่ามากที่สุด กฎเกณฑ์ของความเหมาะสมที่สุดที่นิยมใช้มากที่สุดก็คือ การหาลำดับสถานะที่ดีที่สุดเพียงลำดับเดียว ขั้นตอนวิธีการในการหาลำดับสถานะที่ดีที่สุดเพียงลำดับเดียวนี้อาศัยอยู่บนพื้นฐานของการโปรแกรมแบบพลวัตเรียกว่า "ขั้นตอนวิธีการ Viterbi"

ขั้นตอนวิธีการ Viterbi (Viterbi Algorithm)

ขั้นตอนวิธีการ Viterbi เป็นขั้นตอนวิธีในการหาลำดับสถานะที่ดีที่สุดเพียงลำดับเดียว $Q = \{q_1 q_2 \dots q_T\}$ สำหรับลำดับค่าสังเกตที่กำหนดให้ $O = \{O_1 O_2 \dots O_T\}$ จะกำหนดตัวแปรได้ดังนี้

$$\delta_t(i) = \max_{q_1, q_2, \dots, q_{t-1}} P[q_1 q_2 \dots q_t = i, O_1 O_2 \dots O_t | \lambda] \dots\dots\dots (2.93)$$

โดยที่ $\delta_t(i)$ เป็นค่าความน่าจะเป็นที่มีค่าสูงที่สุดของเส้นทางเดียวที่เวลา t ซึ่งเป็นค่าสังเกต t ค่าแรกและสิ้นสุดในสถานะ S_t ด้วยวิธีการอุปนัยจะได้ว่า

$$\delta_{t+1}(j) = \left[\max_i \delta_t(i) a_{ij} \right] \cdot b_j(O_{t+1}) \dots\dots\dots (2.94)$$

ในการเรียกใช้ค่าลำดับสถานะจำเป็นต้องติดตามค่าอาร์กิวเมนต์ที่ทำให้สมการที่ (2.94) มีค่ามากที่สุดสำหรับแต่ละค่า t และ j โดยอาศัยแถวลำดับ $\psi_t(j)$ ขั้นตอนวิธีการในการหาลำดับสถานะที่ดีที่สุดเพียงลำดับเดียว ดังแสดงในตารางที่ 2.7 รายละเอียดของกระบวนการ Viterbi ดังนี้

ตารางที่ 2.7 รายละเอียดขั้นตอนวิธีการ Viterbi

ขั้นตอนที่ 1	$\delta_1(i) = \pi_i b_i(O_1), \quad 1 \leq i \leq N$	(2.95ก)
กระบวนการเริ่มต้น	$\psi_1(i) = 0$	(2.95ข)
ขั้นตอนที่ 2	$\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}] b_j(O_t), \quad 2 \leq t \leq T$	$1 \leq j \leq N$
กระบวนการวนซ้ำ	$\psi_t(j) = \arg \max_{1 \leq i \leq N} [\delta_{t-1}(i) a_{ij}], \quad 2 \leq t \leq T$	$1 \leq j \leq N$
ขั้นตอนที่ 3	$P^* = \max_{1 \leq i \leq N} [\delta_T(i)]$	(2.97ก)
กระบวนการสิ้นสุด	$q_T^* = \arg \max_{1 \leq i \leq N} [\delta_T(i)]$	(2.97ข)
ขั้นตอนที่ 4	$q_t^* = \psi_{t+1}(q_{t+1}^*),$	(2.98)
กระบวนการย้อนกลับ	$t = T-1, T-2, \dots, 1$	
รอยเส้นทาง		

รายละเอียดขั้นตอนวิธีการ Viterbi นั้นคล้ายคลึงกับการคำนวณไปหน้าของสมการที่ (2.81) ถึง (2.83) ยกเว้นกระบวนการย้อนกลับรอยเส้นทางเท่านั้น ความแตกต่างที่สำคัญก็คือการหาค่าที่สูงที่สุดในสมการที่ (2.96ก) เทียบสถานะก่อนหน้าซึ่งใช้แทนผลรวมในสมการที่ (2.82) นั่นเอง

3.3 การแก้ไขปัญหพื้นฐานข้อที่ 3

ปัญหาพื้นฐานข้อที่สามเกี่ยวข้องกับวิธีการปรับเปลี่ยนพารามิเตอร์ของแบบจำลอง $\lambda = (A, B, \pi)$ เพื่อให้ค่าความน่าจะเป็นของลำดับค่าสังเกตมีค่ามากที่สุดเมื่อกำหนดค่าพารามิเตอร์ของแบบจำลองมาให้ เนื่องจากไม่มีวิธีการที่แน่นอนในการวิเคราะห์เพื่อแก้ไขปัญหสำหรับแบบจำลองที่จะให้ค่าความน่าจะเป็นของลำดับค่าสังเกตมีค่ามากที่สุด ถึงแม้ว่าจะกำหนดลำดับค่าสังเกตที่จำกัดให้เป็นข้อมูลฝึกฝนก็ตามก็ยังไม่มีการใดที่เหมาะสมที่สุดในการประมาณค่าพารามิเตอร์ของแบบจำลอง แต่จะสามารถเลือก $\lambda = (A, B, \pi)$ ที่ทำให้ $P(O|\lambda)$ มีค่ามากที่สุดโดยใช้กระบวนการวนซ้ำของ Baum-Welch

กระบวนการประมาณค่าซ้ำของ Baum-Welch

(Baum-Welch Reestimation Procedure)

กระบวนการประมาณค่าซ้ำของ Baum-Welch นี้อยู่บนพื้นฐานของหลักการของความน่าจะเป็นจริงสูงสุด ซึ่งจะช่วยปรับปรุงให้ค่าความน่าจะเป็นของลำดับค่าสังเกตให้มีค่าสูงขึ้น โดยมีขั้นตอนวิธีการดังนี้ (Picone, 1990)

ตารางที่ 2.8 รายละเอียดกระบวนการประมาณค่าซ้ำของ Baum-Welch

การกระจายของสถานะเริ่มต้น	$\hat{\pi}_i = \alpha_1(i)\beta_1(i) \dots\dots\dots(2.99ก)$
การกระจายของความน่าจะเป็นในการเปลี่ยนแปลงสถานะ	$\hat{a}_{ij} = \frac{\sum_{t=1}^{T-1} \alpha_t(i)a_{ij}b_j(O_{t+1})\beta_{t+1}(j)}{\sum_{t=1}^{T-1} \alpha_t(i)\beta_t(i)} \dots\dots(2.99ข)$
การกระจายของความน่าจะเป็นของสัญลักษณ์ของค่าสังเกต	$\hat{b}_j(k) = \frac{\sum_{t=1, O_t=v_k}^{T-1} \alpha_t(j)\beta_t(j)}{\sum_{t=1}^{T-1} \alpha_t(j)\beta_t(j)} \dots\dots\dots(2.99ค)$

กระบวนการประมาณค่าซ้ำของ Baum-Welch นี้อยู่บนพื้นฐานของความรู้ความเข้าใจในเรื่องของการประมาณค่าใหม่ของความน่าจะเป็นในการเปลี่ยนแปลง ซึ่งอยู่บนพื้นฐานของจำนวนการเปลี่ยนแปลงจากสถานะ i ไปยังสถานะ j ทหารด้วยจำนวนการเปลี่ยนแปลงออกจากสถานะ i ในทำนองเดียวกันค่าความน่าจะเป็นใหม่ของสัญลักษณ์ขาออกสำหรับสัญลักษณ์ที่ k ที่สถานะ i ได้จากจำนวนครั้งของสัญลักษณ์ที่ออกจากสถานะทหารด้วยจำนวนครั้งที่อยู่ในสถานะ

กระบวนการประมาณค่าซ้ำของ Baum-Welch แบบดัดแปลง

(Modified Baum-Welch Reestimation Procedure)

กระบวนการประมาณค่าซ้ำหรือการปรับปรุงและปรับให้ทันกาล ด้วยการวนซ้ำของค่าพารามิเตอร์ของแบบจำลองฮิดเดน มาร์คอฟนั้น เริ่มต้นจากการกำหนดให้ $\xi_t(i, j)$ เป็นค่าความน่าจะเป็นของการอยู่ในสถานะ S_i ที่เวลา t และสถานะ S_j ที่เวลา $t + 1$ เมื่อกำหนดแบบจำลองและลำดับค่าสังเกตให้จะได้ว่า (Rabiner, 1989)

$$\xi_t(i, j) = P(q_t = s_i, q_{t+1} = s_j | O, \lambda) \dots\dots\dots(2.100)$$

ลำดับของเหตุการณ์ที่นำไปสู่เงื่อนไขที่จำเป็นตามสมการที่ (2.98) ดังแสดงในรูปที่ 2.9 จากนิยามของตัวแปรไปหน้าและตัวแปรย้อนกลับจะสามารถเขียน $\xi_r(i, j)$ ด้วยพจน์ของตัวแปรทั้งสองได้ดังสมการที่ (2.94) โดยที่พจน์เศษเป็นเพียง $P(q_r = s_i, q_{r+1} = s_j, O|\lambda)$ ทหารด้วย $P(O|\lambda)$ ซึ่งให้วิธีการวัดค่าความน่าจะเป็นตามที่ต้องการ

$$\begin{aligned} \xi_r(i, j) &= \frac{\alpha_r(i)a_{ij}b_j(O_{r+1})\beta_{r+1}(j)}{P(O|\lambda)} \\ &= \frac{\alpha_r(i)a_{ij}b_j(O_{r+1})\beta_{r+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_r(i)a_{ij}b_j(O_{r+1})\beta_{r+1}(j)} \dots\dots\dots (2.101) \end{aligned}$$

จากการกำหนดให้ค่าของ $\gamma_r(i)$ เป็นค่าความน่าจะเป็นของการอยู่ในสถานะ S_i ที่เวลา t เมื่อกำหนดแบบจำลองและลำดับค่าสังเกตให้ ดังนั้นความสัมพันธ์ระหว่าง $\gamma_r(i)$ และ $\xi_r(i, j)$ เกิดจากผลรวมบน j ทั้งหมดดังนี้

$$\gamma_r(i) = \sum_{j=1}^N \xi_r(i, j) \dots\dots\dots (2.102)$$

ผลรวมของ $\gamma_r(i)$ เหนือตรรกะนี้เวลา t จะเป็นจำนวนครั้งที่เข้าไปยังสถานะ S_i หรือเป็นจำนวนการเปลี่ยนแปลงไปจากสถานะ S_i เมื่อไม่รวมช่วงเวลา $t = T$ จากผลรวม ผลรวมของ $\xi_r(i, j)$ เหนือ t จาก $t = 1$ ถึง $t = T - 1$ จะเป็นจำนวนการเปลี่ยนแปลงจากสถานะ S_i ไปยังสถานะ S_j ดังนี้

$$\sum_{t=1}^{T-1} \gamma_r(i) = \text{จำนวนการเปลี่ยนแปลงไปจากสถานะ } S_i \dots\dots\dots (2.103ก)$$

$$\sum_{t=1}^{T-1} \xi_r(i, j) = \text{จำนวนการเปลี่ยนแปลงไปจากสถานะ } S_i \text{ ไปยังสถานะ } S_j \dots\dots\dots (2.103ข)$$

โดยอาศัยสมการทั้งสองข้างต้นจะได้กระบวนการประมาณค่าซ้ำสำหรับพารามิเตอร์ของแบบจำลองฮิดเดน มาร์คอฟ (A, B, π) ดังแสดงในตารางที่ 2.9 เมื่อกำหนดแบบจำลองด้วย $\lambda = (A, B, \pi)$ เพื่อใช้ในการคำนวณค่าทางด้านขวาของสมการที่ (2.104ก) - (2.104ค) และกำหนดแบบจำลองที่ประมาณค่าใหม่ด้วย $\bar{\lambda} = (\bar{A}, \bar{B}, \bar{\pi})$ ตามค่าทางด้านซ้ายของสมการที่ (2.104ก) - (2.104ค) นอกจากนั้นสมการทั้งสามยังเป็นไปตามเงื่อนไขดังนี้ เงื่อนไขแรก แบบจำลองเริ่มต้น λ เป็นตัวกำหนดจุดวิกฤตของฟังก์ชันความน่าจะเป็นจริงสูงสุดในกรณีนี้ $\bar{\lambda} = \lambda$ เงื่อนไขที่สอง แบบจำลอง $\bar{\lambda}$ มีความน่าจะเป็นจริงสูงกว่าแบบจำลอง λ ด้วยเงื่อนไข $P(O|\bar{\lambda}) > P(O|\lambda)$

ด้วยขั้นตอนพื้นฐานข้างต้นจะกระทำซ้ำโดยนำ $\bar{\lambda}$ มาแทนที่ λ ในการประมาณค่าใหม่ซึ่งเป็นการปรับปรุงค่าความน่าจะเป็นของ O ที่ถูกสังเกตจากแบบจำลองจนกระทั่งถึงจุดสิ้นสุดค่าหนึ่ง ผลลัพธ์ค่าสุดท้ายที่ได้จากการประมาณค่าใหม่เรียกว่า "การประมาณค่าด้วยความน่าจะเป็นจริงสูงสุดของแบบจำลองฮิดเดน มาร์คอฟ" นอกจากนี้แล้วขั้นตอนวิธีการไปหน้า-ย้อนกลับจะเข้าสู่ค่าสูงสุดเฉพาะแห่งเท่า

นั้น ซึ่งเป็นปัญหาที่สำคัญเนื่องจากพื้นผิวของค่าที่เหมาะสมที่สุดมีความสลับซับซ้อนมากและเต็มไปด้วยค่าสูงสุดเฉพาะแห่งมากมาย

ตารางที่ 2.9 รายละเอียดกระบวนการประมาณค่าซ้ำของ Baum-Walch แบบดัดแปลง

การกระจายของสภาวะเริ่มต้น	$\bar{\pi}_i = \gamma_1(i)$(2.104ก)
การกระจายของความน่าจะเป็นในการเปลี่ยนแปลงสถานะ	$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$(2.104ข)
การกระจายของความน่าจะเป็นของสัญลักษณ์ของค่าสังเกต	$\bar{b}_j(k) = \frac{\sum_{t=1}^T \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j)}$(2.104ค)

สมการ (2.104ก) - (2.104ค) ที่ใช้ในการคำนวณสามารถสร้างได้โดยตรงขึ้นมาด้วยการทำให้ฟังก์ชันเสริมของ Baum มีค่าสูงที่สุดเหนือ $\bar{\lambda}$ ดังนี้

$$Q(\lambda, \bar{\lambda}) = \sum_O P(O|\lambda) \log [P(O, \bar{\lambda})] \dots\dots\dots (2.105)$$

ซึ่งการทำให้ค่าของ $Q(\lambda, \bar{\lambda})$ มีค่าสูงที่สุดจะทำให้ความน่าจะเป็นจริงมีค่าสูงขึ้น โดยจะทำให้ฟังก์ชันความน่าจะเป็นจริงลู่เข้าสู่จุดวิกฤตจุดหนึ่งดังนี้

$$\max_{\lambda} [Q(\lambda, \bar{\lambda})] \Rightarrow P(O|\bar{\lambda}) \geq P(O|\lambda) \dots\dots\dots (2.106)$$

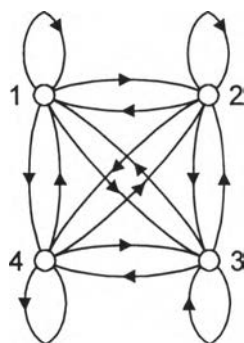
กระบวนการคำนวณค่าใหม่ก็คือขั้นตอนวิธีการ EM ในทางสถิติซึ่งประกอบด้วยสองขั้นตอนย่อย โดยที่ขั้นตอน E (Expectation) หรือการคำนวณค่าคาดหวัง เป็นการคำนวณฟังก์ชันช่วย $Q(\lambda, \bar{\lambda})$ และขั้นตอน M (Modification) หรือการดัดแปลง เป็นการทำให้มีค่าสูงที่สุดเหนือค่าของ $\bar{\lambda}$ นอกจากนี้กระบวนการคำนวณค่าใหม่จะต้องเป็นไปตามเงื่อนไขเฟ้นสุ่ม (Stochastic Constraints) ของพารามิเตอร์ของแบบจำลองฮิดเดน มาร์คอฟ ซึ่งกระบวนการคำนวณค่าใหม่นี้จะเป็นไปตามเงื่อนไขโดยอัตโนมัติดังแสดงในตารางที่ 2.10 รายละเอียดเงื่อนไขเฟ้นสุ่มของพารามิเตอร์ของแบบจำลองฮิดเดน มาร์คอฟ ดังนี้

ตารางที่ 2.10 รายละเอียดเงื่อนไขพื้นฐานของพารามิเตอร์ของแบบจำลองยิตเดน มาร์คอฟ

การกระจายของสภาวะเริ่มต้น	$\sum_{i=1}^N \bar{\pi}_i = 1$(2.107ก)
การกระจายของความน่าจะเป็นในการเปลี่ยนแปลงสถานะ	$\sum_{j=1}^N \bar{a}_{ij} = 1, \quad 1 \leq i \leq N$(2.107ข)
การกระจายของความน่าจะเป็นของสัญลักษณ์ของค่าสังเกต	$\sum_{k=1}^M \bar{b}_j(k) = 1, \quad 1 \leq j \leq N$(2.107ค)

4. ประเภทของแบบจำลองยิตเดน มาร์คอฟ

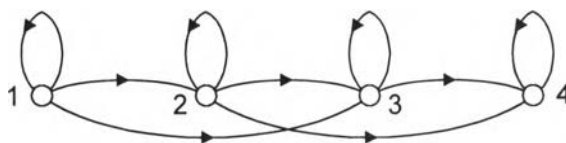
ขั้นตอนวิธีการของแบบจำลองยิตเดน มาร์คอฟส่วนใหญ่เป็นการพิจารณาเพียงกรณีพิเศษของแบบจำลองประเภทเออร์กอดิก ซึ่งเป็นแบบจำลองยิตเดน มาร์คอฟที่ทุกสถานะต่อเชื่อมถึงกันหมด โดยทุกสถานะของแบบจำลองสามารถเข้าถึงสถานะอื่นได้ในขั้นตอนเดียว ดังนั้นแบบจำลองแบบเออร์กอดิกจึงมีคุณสมบัติที่ทุกสถานะสามารถเข้าถึงได้จากสถานะอื่นด้วยขั้นตอนที่จำกัดแน่นอนดังแสดงในรูปที่ 2.11 ซึ่งเป็นแบบจำลองที่มีจำนวนสถานะ $N = 4$ สถานะและมีคุณสมบัติเฉพาะที่สัมพันธ์ a_{ij} ทั้งหมดมีค่าเป็นบวกซึ่งแสดงในสมการที่ (2.108) ดังนี้



รูปที่ 2.11 แบบจำลองแบบเออร์กอดิกที่มี 4 สถานะ

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \dots\dots\dots (2.108)$$

ในการประยุกต์ใช้งานแบบจำลองฮิดเดน มาร์คอฟกับงานเฉพาะอย่างนั้น ยังมีแบบจำลองประเภทอื่นที่เหมาะสมกับคุณสมบัติที่สังเกตของสัญญาณซึ่งถูกจำลองมากกว่าแบบจำลองประเภทเออร์กอดิกมาตรฐานดังแสดงในรูปที่ 2.11 ซึ่งเป็นแบบจำลองซ้าย-ขวา (Left-Right Model) หรือแบบจำลอง Bakis เนื่องจากลำดับสถานะที่อยู่ภายในที่สัมพันธ์กับแบบจำลองมีคุณสมบัติที่ดัชนีของสถานะจะเพิ่มขึ้นหรือมีค่าเท่าเดิมเมื่อเวลาเพิ่มขึ้นเปรียบเสมือนกับสถานะดำเนินจากซ้ายไปขวา ดังนั้นแบบจำลองซ้าย-ขวานี้จึงมีคุณสมบัติเหมาะสมในการจำลองแบบสัญญาณที่เปลี่ยนแปลงไปตามเวลา ดังเช่น สัญญาณเสียง เป็นต้น



รูปที่ 2.12 แบบจำลองแบบซ้าย-ขวาที่มี 4 สถานะ

คุณสมบัติพื้นฐานของทุกแบบจำลองฮิดเดน มาร์คอฟประเภทซ้าย-ขวานั้น สัมประสิทธิ์ของการเปลี่ยนสถานะจะต้องเป็นไปตามคุณสมบัติที่ไม่อนุญาตให้มีการเปลี่ยนแปลงสถานะไปยังสถานะที่มีดัชนีต่ำกว่าสถานะปัจจุบันตามสมการที่ (2.109) รวมทั้งค่าความน่าจะเป็นเริ่มต้นจะต้องมีคุณสมบัติเป็นไปตามสมการที่ (2.110) ดังนี้

$$a_{ij} = 0, \quad j < i \dots\dots\dots (2.109)$$

$$\pi_i = \begin{cases} 0, & i \neq 1 \\ 1, & i = 1 \end{cases} \dots\dots\dots (2.110)$$

เนื่องจากลำดับสถานะจะต้องเริ่มต้นจากสถานะที่ 1 และสิ้นสุดในสถานะที่ N ดังนั้นในแบบจำลองประเภทซ้าย-ขวาจึงต้องเพิ่มเติมเงื่อนไขบังคับให้กับสัมประสิทธิ์ของการเปลี่ยนสถานะ เพื่อไม่ให้เกิดการเปลี่ยนแปลงมากเกินไปดังนี้

$$a_{ij} = 0, \quad j > i + \Delta \dots\dots\dots (2.111)$$

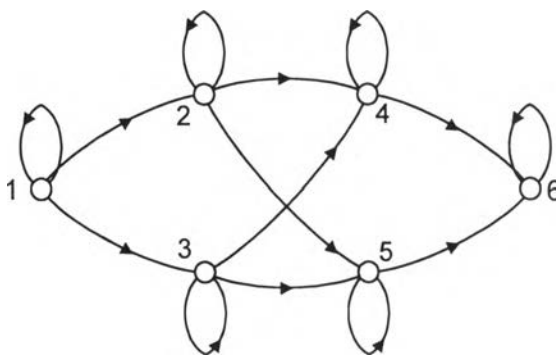
จากรูปที่ 2.12 จะกำหนดให้ค่า $\Delta = 2$ ซึ่งไม่อนุญาตให้มีการข้ามสถานะเกินกว่า 2 สถานะจึงได้เมตริกซ์ของการเปลี่ยนสถานะเป็นดังสมการที่ (2.113) และสถานะสุดท้ายของแบบจำลองประเภทซ้าย-ขวาจะมีสัมประสิทธิ์ของการเปลี่ยนสถานะมีค่าเฉพาะดังสมการที่ (2.112)

$$a_{NN} = 1$$

$$a_{Ni} = 0, \quad i < N \dots\dots\dots (2.112)$$

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & 0 \\ 0 & a_{22} & a_{23} & a_{24} \\ 0 & 0 & a_{33} & a_{34} \\ 0 & 0 & 0 & a_{44} \end{bmatrix} \dots\dots\dots (2.113)$$

นอกจากนั้นยังมีรูปแบบของแบบจำลองที่เป็นไปได้อีกมากมาย ตัวอย่างดังรูปที่ 2.14 ซึ่งแสดงถึงการเชื่อมต่อข้ามแบบจำลองประเภทขนานชาย-ขวาสองชุด แต่แบบจำลองนี้ก็ยังคงจัดอยู่ในแบบจำลองประเภทชาย-ขวาเพียงแต่มีความยืดหยุ่นมากยิ่งขึ้น อย่างไรก็ตามการข้ามสถานะตามเงื่อนไขของแบบจำลองนี้ไม่มีผลต่อกระบวนการประมาณค่าใหม่



รูปที่ 2.13 แบบจำลองแบบเส้นทางขนานชาย-ขวาที่มี 6 สถานะ

จ. ขั้นตอนวิธีการตัดสินใจ (Decision Algorithm)

ขั้นตอนวิธีการตัดสินใจ เป็นขั้นตอนที่เกี่ยวข้องกับกฎเกณฑ์ที่ใช้ในการตัดสินใจเลือกรูปแบบที่มีความคล้ายคลึงกันมากที่สุดระหว่างคำพูดที่ไม่ทราบรูปแบบกับรูปแบบที่ได้จัดเก็บไว้ล่วงหน้า โดยอาศัยค่าความไม่คล้ายคลึงกันหรือค่าระยะทางที่ได้จากการทดสอบความคล้ายคลึงกันของรูปแบบที่มีค่ามากที่สุด ในขั้นตอนวิธีการแก้ปัญหาพื้นฐานทั้งสามประการของแบบจำลองฮิดเดน มาร์คอฟนั้น การแก้ปัญหาพื้นฐานข้อที่สองจัดเป็นขั้นตอนวิธีการตัดสินใจเลือกรูปแบบที่เหมาะสมที่สุดในการรู้จำ วิธีการหนึ่งที่ถูกนำมาใช้กับแบบจำลองฮิดเดน มาร์คอฟก็คือขั้นตอนวิธีการของ Viterbi (Rabiner and Levinson, 1981; Rabiner, 1989) โดยมีขั้นตอนวิธีการดังแสดงในการแก้ปัญหาพื้นฐานข้อที่สอง ส่วนรายละเอียดเชิงทฤษฎีมีดังนี้

รายละเอียดเชิงทฤษฎีของขั้นตอนวิธีการของ Viterbi

ขั้นตอนวิธีการของ Viterbi เป็นการแก้ปัญหาโดยการวนซ้ำเพื่อหาคำตอบที่เหมาะสมที่สุดของปัญหาในการประมาณค่าลำดับสถานะ หรือปัญหาในการประมาณค่าความน่าจะเป็นสูงสุดโดยการอุปนัย (Maximum a posteriori Probability, MAP) สำหรับกระบวนการมาร์คอฟที่มีสถานะจำกัดและเวลาไม่ต่อเนื่อง (Forney, Jr., 1973)

ภายใต้กระบวนการมาร์คอฟโดยเวลาไม่ต่อเนื่อง สถานะ x_k ที่เวลา k ซึ่งมีจำนวนจำกัด M ของสถานะ $m, 1 \leq m \leq M$ ขั้นแรกเริ่มต้นจากการสมมติให้กระบวนการอยู่ในช่วงเวลา 0 ถึง K เท่านั้นโดยมีสถานะเริ่มต้นเป็น x_0 และสถานะสิ้นสุดเป็น x_K ลำดับ สถานะจะแทนได้ด้วยเวกเตอร์จำกัด

$\mathbf{x} = (x_0, \dots, x_k)$ ดังนั้นกระบวนการมาร์คอฟซึ่งมีความน่าจะเป็น $P(x_{k+1}|x_0, x_1, \dots, x_k)$ ของการอยู่ในสถานะ x_{k+1} ที่เวลา $k+1$ จะให้สถานะทั้งหมดจนถึงเวลา k โดยขึ้นอยู่กับสถานะ x_k ที่เวลา k เท่านั้น โดยที่ค่าความน่าจะเป็นของการเปลี่ยนแปลง $P(x_{k+1}|x_k)$ อาจขึ้นกับเวลาดังนี้

$$P(x_{k+1}|x_0, x_1, \dots, x_k) = P(x_{k+1}|x_k) \dots \dots \dots (2.114)$$

กำหนดให้การเปลี่ยนแปลง ξ_k ที่เวลา k เป็นคู่สถานะ (x_{k+1}, x_k) ดังนี้

$$\xi_k \equiv (x_{k+1}, x_k) \dots \dots \dots (2.115)$$

กำหนดให้ Ξ เป็นชุดของการเปลี่ยนแปลง $\xi_k = (x_{k+1}, x_k)$ ซึ่ง $P(x_{k+1}|x_k) \neq 0$ และมีค่าเป็น $|\Xi|$ โดยที่ $|\Xi| \leq M^2$ ดังนั้นจึงเป็นความสัมพันธ์แบบหนึ่งต่อหนึ่งระหว่างลำดับสถานะ \mathbf{x} และลำดับการเปลี่ยนแปลง $\xi = (\xi_0, \dots, \xi_{k-1})$ ซึ่งเขียนได้เป็น $\mathbf{x} \leftrightarrow \xi$ กระบวนการได้รับการสมมติให้ถูกสังเกต โดยประกอบไปด้วยลำดับ z ของค่าสังเกต z_k ซึ่งขึ้นอยู่กับค่าการเปลี่ยนแปลง ξ_k ที่เวลา k ในเชิงความน่าจะเป็นดังนี้

$$P(z|x) = P(z|\xi) = \prod_{k=0}^{k-1} P(z_k|\xi_k) \dots \dots \dots (2.116)$$

ในกรณีที่มีการเปลี่ยนแปลงตามเวลาซึ่ง $P(z_k|\xi_k)$ เป็นฟังก์ชันของ k เป็นกรณีพิเศษคือ

1) กรณีที่ z_k ขึ้นอยู่กับสถานะ x_k เท่านั้น

$$P(z|x) = \prod_k P(z_k|\xi_k) \dots \dots \dots (2.117)$$

2) กรณีที่ z_k ขึ้นอยู่กับขาออก y_k ของกระบวนการในเชิงความน่าจะเป็น ซึ่ง y_k เป็นฟังก์ชันที่สามารถหาค่าได้ของค่าการเปลี่ยนแปลง ξ_k หรือสถานะ x_k

หลักการของขั้นตอนวิธีการของ Viterbi

ขั้นตอนวิธีการของ Viterbi เป็นการหาค่าตอบในการประมาณลำดับค่าความน่าจะเป็นสูงสุดโดยการอุปนัย ซึ่งเป็นวิธีเดียวกับการหาเส้นทางที่สั้นที่สุดในแผนภูมิ จากรูปที่ 2.14 เป็นแผนภาพสถานะของกระบวนการมาร์คอฟที่มีสถานะจำกัดและเวลาไม่ต่อเนื่อง โดยอาศัยปมแทนสถานะและกิ่งก้านสาขาแทนการเปลี่ยนแปลง ซึ่งภายในช่วงเวลาหนึ่งกระบวนการจะดำเนินไปตามเส้นทางจากสถานะหนึ่งไปยังอีกสถานะหนึ่งผ่านทางแผนภาพสถานะ

จากรูปที่ 2.15 แสดงแผนภาพ Trellis ซึ่งเป็นรายละเอียดของกระบวนการเดียวกันที่ซ้ำซ้อนมากยิ่งขึ้นโดยอาศัยปมแทนสถานะที่เวลาที่กำหนด และแต่ละกิ่งก้านสาขาแทนการเปลี่ยนแปลงไปยังสถานะใหม่ในเวลาถัดไป แผนภาพ Trellis จะเริ่มต้นและสิ้นสุดลงที่สถานะ x_0 และ x_k ตามลำดับ โดยมีคุณสมบัติที่สำคัญซึ่งทุกลำดับสถานะ \mathbf{x} ที่เป็นไปได้ จะสัมพันธ์กับเส้นทางเอกลักษณ์ที่มีเพียงเส้นทางเดียวในแผนภาพ Trellis และเป็นจริงในทางกลับกัน

เมื่อกำหนดลำดับของค่าสังเกต \mathbf{x} มาให้ ทุกเส้นทางอาจถูกกำหนดความยาวให้เป็นสัดส่วนกับ $-\ln P(\mathbf{x}, \mathbf{z})$ เมื่อ \mathbf{x} เป็นลำดับสถานะที่สัมพันธ์กับเส้นทางนั้น ซึ่งช่วยในการค้นหาลำดับสถานะซึ่ง $P(\mathbf{x}|\mathbf{z})$ มีค่าสูงสุดหรืออีกนัยหนึ่ง $P(\mathbf{x}, \mathbf{z}) = P(\mathbf{x}|\mathbf{z})P(\mathbf{z})$ มีค่าสูงสุด โดยการค้นหาเส้นทางที่ความยาว $-\ln P(\mathbf{x}, \mathbf{z})$ มีค่าน้อยที่สุด เนื่องจาก $\ln P(\mathbf{x}, \mathbf{z})$ เป็นฟังก์ชันแบบไม่เปลี่ยนแปลงของ $P(\mathbf{x}, \mathbf{z})$ ซึ่งมีความสัมพันธ์แบบหนึ่งต่อหนึ่งระหว่างเส้นทางและลำดับ ดังนั้นจึงสามารถสังเกต $P(\mathbf{x}, \mathbf{z})$ ได้เป็นดังนี้

$$P(\mathbf{x}, \mathbf{z}) = P(\mathbf{x})P(\mathbf{z}|\mathbf{x}) \\ = \prod_{k=0}^{K-1} P(x_{k+1}|x_k) \prod_{k=0}^{K-1} P(z_k|x_{k+1}, x_k) \quad (2.118)$$

ดังนั้นเมื่อกำหนดแต่ละกิ่งสาขาของการเปลี่ยนแปลงด้วยความยาวจะได้ว่า

$$\lambda(\xi_k) \cong -\ln P(x_{k+1}|x_k) - \ln P(z_k|\xi_k) \quad (2.119)$$

ดังนั้นความยาวทั้งหมดของเส้นทางที่สัมพันธ์กับ \mathbf{x} บางตัวจะมีค่าดังนี้

$$-\ln P(\mathbf{x}, \mathbf{z}) = \sum_{k=0}^{K-1} \lambda(\xi_k) \quad (2.120)$$

ในการค้นหาเส้นทางที่สั้นที่สุดบนแผนภูมินั้น มีพื้นฐานจากวิธีการของ Minty ซึ่งต้องการค่าสังเกตเพิ่มเติมอีกหนึ่งค่า เริ่มจากการกำหนดให้ \mathbf{x}_0^k เป็นส่วนย่อยของ (x_0, x_1, \dots, x_k) ที่ประกอบด้วยลำดับที่เวลา k ของลำดับสถานะ $\mathbf{x} = (x_0, x_1, \dots, x_k)$ ในแผนภาพ Trellis ที่ \mathbf{x}_0^k สัมพันธ์กับส่วนย่อยของเส้นทางที่เริ่มต้นจากปม \mathbf{x}_0 และสิ้นสุดที่ปม \mathbf{x}_k ดังนั้นสำหรับปม \mathbf{x}_k ที่เฉพาะเวลา k ใดๆ จะประกอบด้วยส่วนย่อยของเส้นทางดังกล่าวหลายส่วนซึ่งมีความยาวเป็นดังนี้

$$\lambda(\mathbf{x}_0^k) = \sum_{i=0}^{k-1} \lambda(\xi_i) \quad (2.121)$$

ส่วนของเส้นทางที่สั้นที่สุดจะสัมพันธ์กับปม x_k และแทนด้วย $\hat{\mathbf{x}}(x_k)$ ที่เวลา $k > 0$ ใดๆ จะประกอบด้วย M เส้นทางสำหรับแต่ละ x_k โดยที่ค่าสังเกตจะเป็นเส้นทางสมบูรณ์ที่สั้นที่สุด $\hat{\mathbf{x}}$ ซึ่งเริ่มต้นจากส่วนของเส้นทางที่สั้นที่สุด ในทางกลับกันถ้าผ่านทางสถานะ x_k ที่เวลา k จะแทนที่ส่วนของเส้นทางเริ่มต้นด้วย $\hat{\mathbf{x}}(x_k)$ ซึ่งกลายเป็นเส้นทางที่สั้นที่สุด

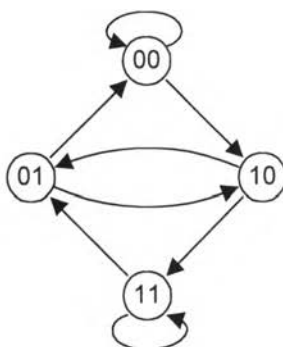
ดังนั้นที่เวลา k ใดๆ จำเป็นต้องบันทึกไว้แต่เพียงเส้นทาง M เส้นทาง $\hat{\mathbf{x}}(x_k)$ พร้อมทั้งความยาว $\Gamma(x_k) \cong \lambda[\hat{\mathbf{x}}(x_k)]$ ของเส้นทางนั้น ที่เวลา $k + 1$ จะอาศัยการยืดขยายเส้นทางทั้งหมดที่เวลา k ออกไปหนึ่งหน่วยเวลา คำนวณหาความยาวในส่วนของเส้นทางที่ยืดขยายออกมา สำหรับแต่ละปม x_{k+1} จะเลือกส่วนของเส้นทางที่ยืดขยายออกมาที่สั้นที่สุดซึ่งสิ้นสุดที่ x_{k+1} โดยสัมพันธ์กับเส้นทางที่เวลา $k + 1$ กระบวนการวนซ้ำจะกระทำต่อไปจนกว่าจำนวนเส้นทางจะเกินกว่า M

ตัวอย่างของขั้นตอนวิธีการ Viterbi ดังแสดงในรูปที่ 2.14 ด้วยแผนภาพ Trellis 4 สถานะที่ประกอบด้วย 5 หน่วยเวลา โดยบอกความยาวของแต่ละกิ่งสาขาไว้ด้วย ในรูปที่ 2.15 เป็นกระบวนการวนซ้ำ 5 รอบเพื่อค้นหาเส้นทางที่สั้นที่สุดจากปมเริ่มต้นไปยังปมสุดท้าย ในแต่ละขั้นจะแสดงเพียง 4 เส้นทางเท่านั้น

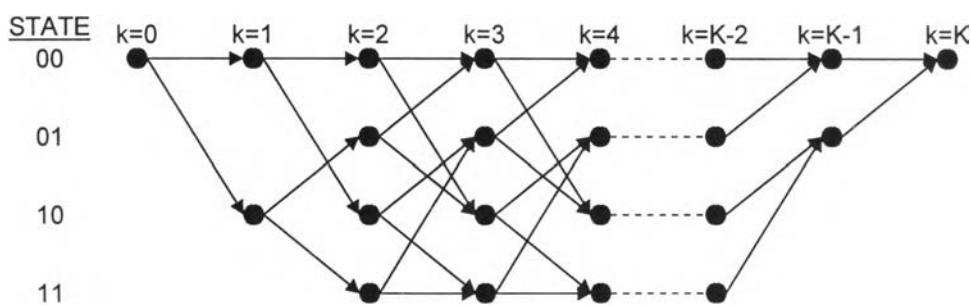
พร้อมทั้งความยาวของแต่ละเส้นทาง ดังนั้นจากหลักการของขั้นตอนวิธีการดังแสดงในตารางที่ 2.11 จะได้ลำดับสถานะ x ที่มีจำนวนจำกัดซึ่งสิ้นสุดที่เวลา K ร่วมกับเส้นทางสมบูรณ์ที่สั้นที่สุด $\hat{x}(x_K)$

ตารางที่ 2.11 หลักการเชิงทฤษฎีของขั้นตอนวิธีการ Viterbi

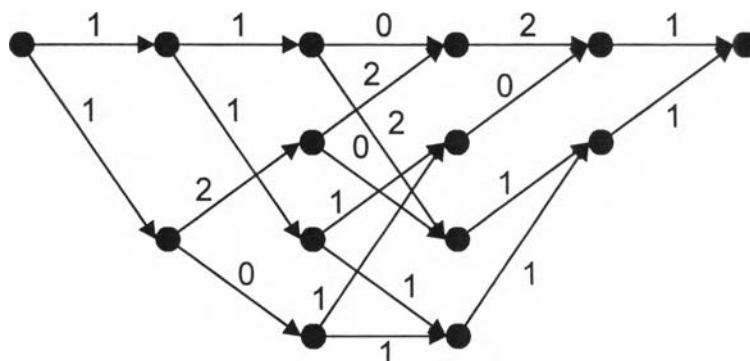
ขั้นตอนที่ 1	k	ดัชนีเวลา
กระบวนการกำหนด	$\hat{x}_k(x_k), 1 \leq x_k \leq M$	เส้นทางที่สั้นที่สุดที่ x_k
ค่าเริ่มต้น	$\Gamma(x_k), 1 \leq x_k \leq M$	ความยาวของเส้นทาง
ขั้นตอนที่ 2	$k = 0$	(2.122ก)
กระบวนการเริ่มต้น	$\hat{x}(x_0) = x_0, \hat{x}(m)$ arbitrary, $m \neq x_0$	(2.122ข)
	$\Gamma(x_0) = 0, \Gamma(m) = \infty, m \neq x_0$	(2.122ค)
ขั้นตอนที่ 3	$\Gamma(x_{k+1}, x_k) \equiv \Gamma(x_k) + \lambda[\xi_k = (x_{k+1}, x_k)],$	(2.123ก)
กระบวนการวนซ้ำ	$\forall \xi_k = (x_{k+1}, x_k)$	
	$\Gamma(x_{k+1}) = \min_{x_k} \Gamma(x_{k+1}, x_k),$ สำหรับแต่ละ x_{k+1}	(2.123ข)
ขั้นตอนที่ 4	จัดเก็บ $\Gamma(x_{k+1})$ และเส้นทาง $\hat{x}(x_{k+1})$ ที่สัมพันธ์กัน	
กระบวนการสิ้นสุด	กำหนดให้ $k = k + 1$ และวนซ้ำจะกระทำ $k = K$	



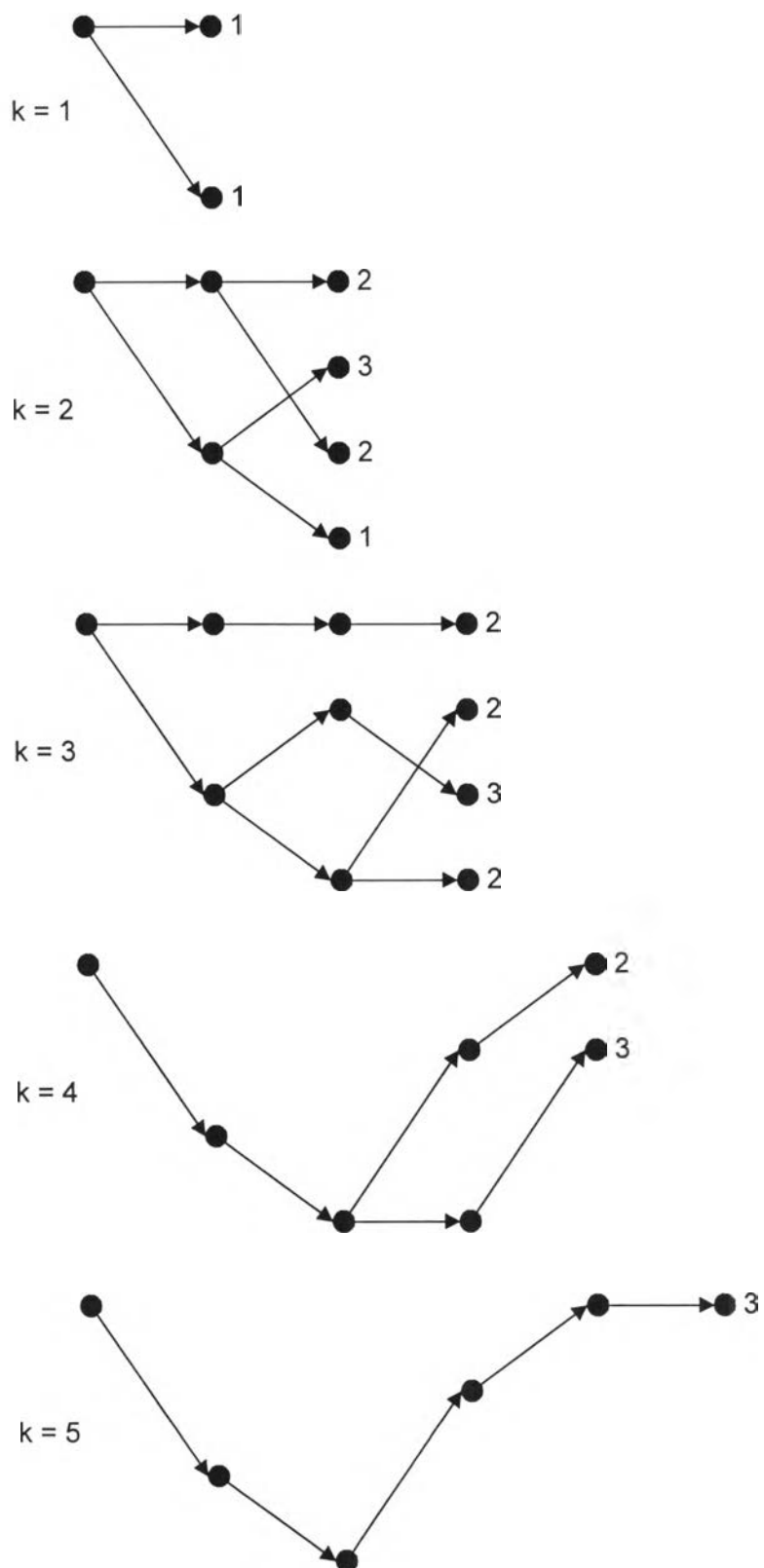
รูปที่ 2.14 แผนภาพสถานะ



รูปที่ 2.15 แผนภาพ Trellis



รูปที่ 2.16 แผนภาพ Trellis พร้อมแสดงขนาดความยาวของกิ่งสาขาเมื่อ $M = 4$ และ $K = 5$



รูปที่ 2.17 กระบวนการค้นหาเส้นทางที่สั้นที่สุดตามหลักการของขั้นตอนวิธีการ Viterbi