

โปรแกรมการลดขนาดข้อมูลเสียง
PROGRAM COMPRESSION OF SPEECH

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

รายงานวิจัย



สำนักวิจัยและบริการคอมพิวเตอร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

RCH

QA

Y6.9

D33

๗๖๒๙

ปีงบประมาณ 2541

เลขหมู่.....
เลขทะเบียน.....
วัน, เดือน, ปี.....

35259

๒๕๔๑

๒๕๔๑

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อให้บริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
โดยไม่ได้รับอนุญาตจากสำนักหอสมุดกลาง หากมีข้อสงสัยหรือต้องการข้อมูลเพิ่มเติม กรุณาติดต่อสำนักหอสมุดกลาง โทร. ๐๒-๕๑๖๒๙๑

สารบัญ

	หน้า
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาของโครงการลดขนาดข้อมูลเสียงภาษาไทย.....	1
1.2 จุดประสงค์ของงานวิจัย.....	2
1.3 ขอบเขตของงานวิจัย.....	2
1.4 วิธีการดำเนินงาน.....	2
บทที่ 2 ลักษณะของสัญญาณเสียง และการประมวลผล.....	3
2.1 ลักษณะของข้อมูลเสียง.....	3
2.2 ระบบการวิเคราะห์เสียงในรูปแบบต่างๆ.....	4
บทที่ 3 การแปลงข้อมูลเสียงให้อยู่ในรูปของสเปกโตรแกรม.....	5
บทที่ 4 ทฤษฎีการแปลงข้อมูลแบบคาร์ฮูเนนเลิฟ.....	8
4.1 คณิตศาสตร์การแปลงข้อมูลแบบคาร์ฮูเนนเลิฟ.....	8
4.2 คำอธิบายเสริมแบบจำลองทางคณิตศาสตร์.....	11
บทที่ 5 ความรู้เบื้องต้นเกี่ยวกับโครงข่ายประสาทเทียม.....	14
5.1 โครงข่ายประสาทเทียม.....	15
5.2 ฟังก์ชันกระตุ้นความสนใจ.....	16
5.3 โครงข่ายประสาทเทียมแบบชั้นเดียว.....	18
5.4 โครงข่ายประสาทเทียมแบบหลายชั้น.....	20
5.5 การฝึกสอนให้กับโครงข่ายประสาทเทียม.....	20
5.6 วิธีการแก้ปัญหาการฝึกสอน.....	21
5.7 เงื่อนไขการหยุดการฝึกสอน.....	24
บทที่ 6 ขั้นตอนการทดลอง.....	25
6.1 ประยุกต์การแปลงข้อมูลแบบคาร์ฮูเนนเลิฟ(Karhunen-Loeve Transform) ใช้ในการลดขนาดข้อมูลเสียง.....	25
6.2 การประยุกต์ใช้วิธีการแปลงของคาร์ฮูเนนเลิฟกับข้อมูลเสียงในแกนเวลา.....	26
6.3 การประยุกต์ใช้วิธีการแปลงของคาร์ฮูเนนเลิฟกับข้อมูลเสียงในแกนความถี่.....	27
6.4 การนำผลลัพธ์ที่ได้จากการแปลงข้อมูลแบบคาร์ฮูเนนเลิฟ เข้าสู่ขั้นตอน การเรียนรู้โดยใช้โครงข่ายประสาทเทียม.....	28

บทที่ 1

บทนำ

1.1 ความเป็นมาของโครงการลดขนาดข้อมูลเสียงภาษาไทย

ปัจจุบันนี้ เครือข่ายการสื่อสาร ได้มีการขยายตัวเชื่อมโยงไปทั่วโลกตามกระแสโลกาภิวัตน์ ปริมาณข่าวสารที่ถูกส่งไปยังเครือข่ายต่างๆ ก็เพิ่มขึ้นอย่างรวดเร็ว ในจำนวนข่าวสารที่ถูกส่งไปนี้ก็ได้รวมถึงข่าวสารเสียงพูดด้วย

ที่ผ่านมามการส่งผ่านข่าวสารเสียงพูดผ่านเครือข่ายเชิงเลขที่ส่งไปโดยไม่มีกรลดขนาดข้อมูลเสียงพูดจะถูกส่งด้วย อัตรา 8,000 ตัวอย่างต่อวินาที โดยแต่ละตัวมีขนาด 8 bit นั่นคือ bit-rate ที่ต้องส่งไปในเครือข่ายเป็น 64,000 bps ซึ่งในความเป็นจริงแล้วถ้าเราสามารถลดขนาดข้อมูลเสียงพูดลงไปได้ เราจะสามารถส่งข่าวสารผ่านเครือข่าย ได้มากขึ้นในเวลาเท่ากัน โดยไม่ต้องเปลี่ยนแปลงความจุของเครือข่ายการสื่อสาร

นอกจากการบีบข้อมูลเสียงจะมีประโยชน์ในการลดขนาดข้อมูลข่าวสารผ่านเครือข่ายแล้ว ยังสามารถเป็นการลดขนาดข้อมูลสำหรับการประมวลผลในงานทางด้านอื่นอีก เช่น เป็นการลดขนาดข้อมูลสำหรับการรู้จำเสียง ในงานวิจัยนี้ได้เน้นประโยชน์ของการบีบข้อมูลเพื่อเป็นการลดขนาดของข้อมูลเสียงสำหรับที่จะนำไปประมวลผลในด้านรู้จำเสียง หมายถึง นอกจากพารามิเตอร์ต่างๆ ที่เราได้จากการบีบข้อมูลจะสามารถสร้างข้อมูลเดิมกลับมาได้ แล้วยังสามารถนำพารามิเตอร์นั้นเป็นตัวแทนของข้อมูลเสียงสำหรับเข้าสู่ขั้นตอนการรู้จำได้อีก

ซึ่งปัญหาที่สำคัญที่พบในงานวิจัยเกี่ยวกับการรู้จำเสียงพูดก็คือ ขนาดของข้อมูลเสียงที่ใหญ่ มากซึ่งมีผลต่อขนาดเนื้อที่ในหน่วยความจำและเวลาที่ใช้ในการประมวลผล ถึงแม้ว่าเราจะพยายามนำเอาเทคนิคทางด้านโครงข่ายประสาทเทียม (Artificial Neural Network) ซึ่งมีประสิทธิภาพอย่างยิ่งในการลดเวลาที่ใช้ในการประมวลผลข้อมูลมาประยุกต์ใช้

ในรายงานนี้จะนำเสนอวิธีการลดขนาดของข้อมูลเสียงโดยใช้วิธีการแปลงที่เรียกว่าการแปลงของคาร์สุเนนลิป ซึ่งเป็นวิธีที่งานวิจัยในต่างประเทศ ที่เกี่ยวกับการประมวลผลข้อมูลเสียง และการรู้จำรูปแบบโดยทั่วไป นำมาประยุกต์ใช้ [1] วิธีการนี้สามารถดึงเอาลักษณะเฉพาะ ของข้อมูลเสียงออกมาได้ ซึ่งจะแทนด้วยเวกเตอร์โอเก็นและเวกเตอร์ตี้มประสิทธิ์ ที่มีผลรวมของค่า โอเก็นประมาณร้อยละ 95 ของค่าโอเก็นทั้งหมด ซึ่งสามารถลดขนาดของ ข้อมูลได้ถึงร้อยละ 40-50 ของข้อมูลเสียง นอกจากนี้งานวิจัยยังนำเสนอผลการ เปรียบเทียบระหว่างการนำเอาวิธี การแปลงของคาร์สุเนนลิปมาประยุกต์ใช้ สำหรับลดขนาดข้อมูลทั้งในแง่เวลา และแแกนความถี่

1.2 จุดประสงค์ของงานวิจัย

1. เพื่อศึกษา การแปลงข้อมูลแบบคาร์สุเนนเลิฟมาใช้ในการประมวลผล
2. เพื่อประยุกต์ การแปลงข้อมูลแบบคาร์สุเนนเลิฟ เพื่อทำการลดขนาดข้อมูลเสียงพูดภาษาไทย
3. เพื่อศึกษา ผลลัพธ์ที่ได้จากการลดขนาดข้อมูลนำไปใช้ประโยชน์ในการรู้จำเสียงภาษาไทย

1.3 ขอบเขตของงานวิจัย

1. พัฒนาโปรแกรมลดขนาดข้อมูลเสียงโดยประยุกต์ใช้การแปลงข้อมูลแบบคาร์สุเนนเลิฟ
2. ทำการทดลองลดขนาดข้อมูลเสียงทั้งในแกนเวลา และแกนความถี่
3. เปรียบเทียบผลของการลดขนาดข้อมูลเสียงในแกนเวลา และแกนความถี่ โดยเทียบกับ อัตราการลดขนาดข้อมูล กับคุณภาพของข้อมูลเสียงที่แปลงกลับคืนมา
4. นำผลลัพธ์ที่ได้จากการลดขนาดข้อมูล เข้าสู่ขั้นตอนการเรียนรู้ และการรู้จำเสียงภาษาไทย

1.4 วิธีการดำเนินงาน

1. ศึกษาลักษณะของสัญญาณเสียงโดยทั่วไป
2. ศึกษาเรียนรู้ทฤษฎีการแปลงข้อมูลจากแกนเวลา เป็นแกนความถี่
3. เรียนรู้ทฤษฎี และการประยุกต์ใช้การแปลงข้อมูลแบบคาร์สุเนนเลิฟ
4. เขียนโปรแกรมลดขนาดข้อมูลเสียงทั้งในแกนเวลา และแกนความถี่
5. ทดลอง และวัดประสิทธิภาพของการลดขนาดข้อมูล โดยเทียบกันระหว่างการลดขนาดข้อมูลในแกนเวลา และแกนความถี่
6. นำผลลัพธ์ที่ได้จากการลดขนาดข้อมูล เข้าสู่ขั้นตอนการรู้จำโดยใช้โครงข่ายประสาทเทียม

บทที่ 2

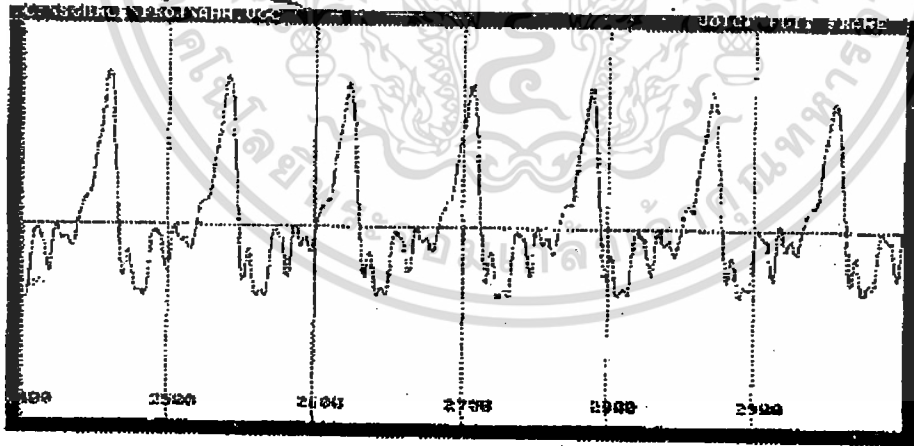
ลักษณะของสัญญาณเสียง และการประมวลผล

2.1 ลักษณะของข้อมูลเสียง

ในขั้นตอนแรกนี้ เป็นการแปลงสัญญาณเสียงจากผู้พูด ให้เป็นกลุ่มข้อมูลดิจิทัลโดยได้เลือกใช้การ์ด Sound Blaster Pro ในการสุ่มสัญญาณ โดยภายในการ์ดจะมีวงจรกรองความถี่ต่ำผ่าน (Low-pass Filter) ในตัว ในการทดลอง ได้กำหนดความถี่สุ่มเท่ากับ 10 กิโลเฮิร์ตซ์เนื่องจากเสียงพูดตามปกติของมนุษย์ จะมีรายละเอียดส่วนใหญ่อยู่ในย่านความถี่ไม่เกิน 5 กิโลเฮิร์ตซ์ ความถี่สุ่มนี้อาจปรับได้สูงกว่านี้ได้ แต่ข้อมูลที่ได้อาจไม่จำเป็นต่อการวิเคราะห์และรู้จำ ข้อมูลที่ได้จากการสุ่มนี้จะเป็นค่าปริมาณเสียงในแต่ละคาบเวลา ดังสมการที่ 2.1

$$V(nT) = V(t) \delta(nT) \dots\dots 2.1$$

เมื่อ $V(t)$ เป็นสัญญาณเสียงในรูปของสัญญาณอนาล็อก
 $V(nT)$ เป็นสัญญาณเสียงที่ได้จากการสุ่ม
 T เป็นคาบเวลาที่ใช้ในการสุ่มในที่นี้มีค่าเท่ากับ 100 ไมโครวินาที
 n เลขจำนวนเต็มบวก



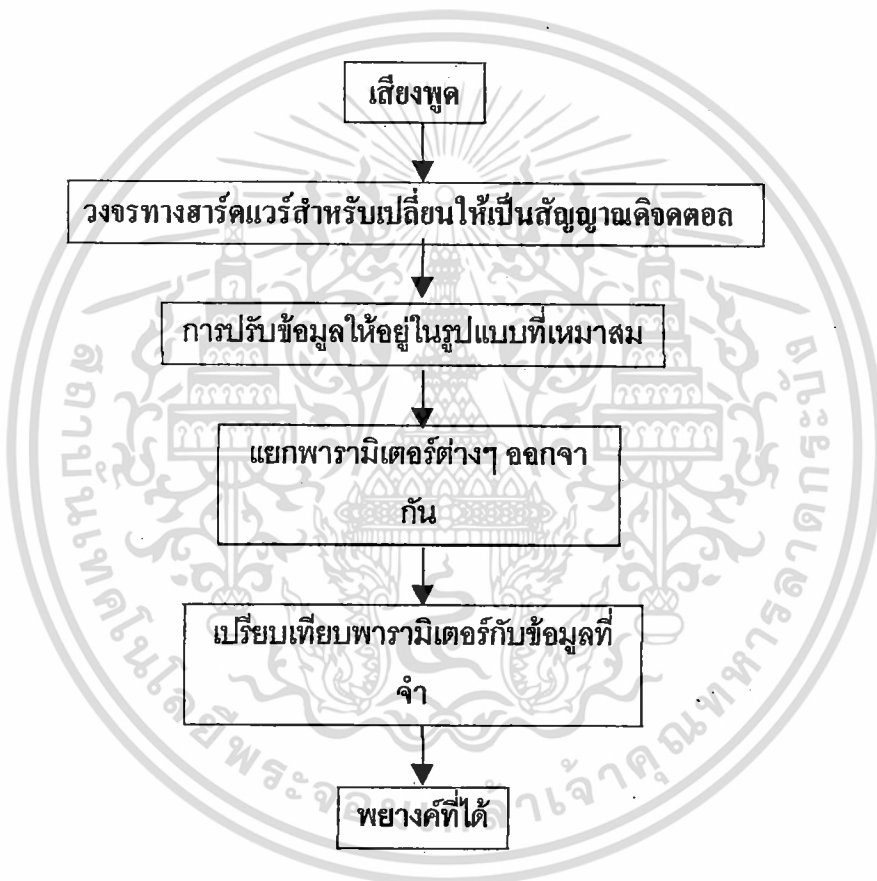
รูปที่ 2.1 สัญญาณตัวอย่างที่ได้จากการสุ่ม

จากรูปที่ 2.1 เป็นรูปสัญญาณเสียงที่สุ่มมาได้จากโปรแกรมที่พัฒนาขึ้น สัญญาณดังกล่าวในแต่ละจุดของการสุ่มจะใช้พื้นที่เก็บข้อมูล 8 บิต จึงจำลองสัญญาณได้ 256 ระดับ ในบางกรณีนี้สัญญาณมีความอ่อนมาก เราอาจคูณสัญญาณด้วยค่าคงที่ เพื่อให้ได้สัญญาณที่แรงขึ้นก่อนที่จะนำไปคำนวณต่อไป

2.2 ระบบการวิเคราะห์เสียงในรูปแบบต่างๆ

ในปัจจุบันอุปกรณ์ที่มีใช้ในชีวิตประจำวันมีส่วนประกอบของคอมพิวเตอร์มากขึ้นทำให้อุปกรณ์ต่างๆ เหล่านี้สามารถทำงานได้ตามโปรแกรมที่วางไว้ล่วงหน้า หรือสามารถตอบสนองคำสั่งของมนุษย์โดยการรับการกดปุ่ม หรือส่งสัญญาณข้อมูลจากระบบสื่อสารต่างๆ ได้

นอกจากการสั่งงานอุปกรณ์ด้วยการกดปุ่ม หรือใช้การสื่อสารผ่านทางสื่อต่างๆ แล้วยังมีความพยายามที่จะใช้สัญญาณเสียง ในการสั่งงานให้คอมพิวเตอร์ที่อยู่ในอุปกรณ์เหล่านั้นให้ทำงานตามที่ต้องการได้ อุปกรณ์เหล่านี้จึงต้องมีส่วนการทำงานด้านการรู้จำเสียง (Recognize) จากผู้พูด การรู้จำเสียงในลักษณะนี้มีกรรมวิธีโดยคร่าวๆ ดังรูปที่ 2.2



รูปที่ 2.2 ระบบการรู้จำ

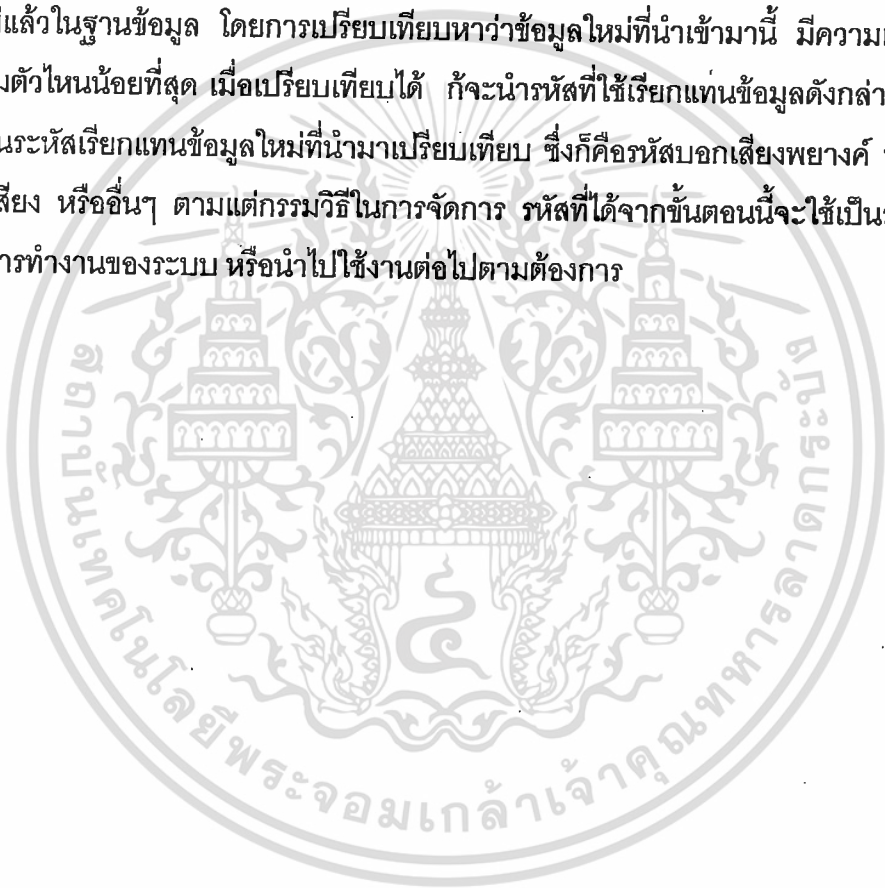
การรู้จำเสียงเพื่อนำไปแปลงเป็นสัญญาณในการควบคุม อาจแบ่งเป็นขั้นตอนอย่างคร่าวๆ ได้ดังนี้

1. แปลงสัญญาณเสียงให้เป็นสัญญาณดิจิทัล สัญญาณเสียงจะถูกบันทึกไว้โดยไมโครโฟน สัญญาณที่ได้ อาจถูกนำมาผ่านวงจรของความถี่ต่ำ (Low-pass Filter) และนำมาสุ่ม (Sampling) เพื่อให้ได้สัญญาณดิจิทัล การผ่านวงจรความถี่ต่ำก่อนที่จะสุ่มสัญญาณ เพื่อป้องกันกาเกิด Aliasing สัญญาณที่ผ่านจากขั้นตอนนี้จะเป็นกลุ่มข้อมูลของระดับสัญญาณในแกนของเวลา

2. การปรับข้อมูลให้อยู่ในรูปแบบที่เหมาะสม โดยการแปลง (transform) สัญญาณที่ได้จากขั้นตอนที่แล้วให้อยู่ในโดเมนที่เหมาะสม เช่นการแปลงฟูรีเยร์ (Fourier Transform) ,การแปลงแบบดิสครีตโคร์ซาย (DCT) ข้อมูลที่ผ่านขั้นตอนี้จะมีความแตกต่างไปตามตัวผู้พูด หรือมีความแตกต่างไปตามเสียงพยางค์ที่พูด เป็นต้น

3. แยกพารามิเตอร์ออกจากกัน โดยการนำข้อมูลที่ได้จากขั้นตอนที่สอง มาแยกพารามิเตอร์ที่ต้องการ เพื่อลดขนาดข้อมูลที่ต้องการจัดการ แต่ในบางวิธี อาจใช้ข้อมูลทั้งหมดที่ได้จากขั้นตอนที่สองเลยก็ได้

4. นำพารามิเตอร์ที่ได้จากขั้นตอนที่สาม หรือข้อมูลจากขั้นตอนที่สอง มาเปรียบเทียบกับข้อมูลเดิมที่มีอยู่แล้วในฐานข้อมูล โดยการเปรียบเทียบหาว่าข้อมูลใหม่ที่น่าเข้ามา นี้ มีความแตกต่างกับข้อมูลเดิมตัวไหนน้อยที่สุด เมื่อเปรียบเทียบได้ ก็จะมีรหัสที่ใช้เรียกแทนข้อมูลดังกล่าวในฐานข้อมูลมาเป็นรหัสเรียกแทนข้อมูลใหม่ที่น่ามาเปรียบเทียบ ซึ่งก็คือรหัสบอกเสียงพยางค์ หรือรหัสบอกหน่วยเสียง หรืออื่นๆ ตามแต่กรรมวิธีในการจัดการ รหัสที่ได้จากขั้นตอนนี้จะใช้เป็นรหัสในการควบคุมการทำงานของระบบ หรือนำไปใช้งานต่อไปตามต้องการ



บทที่ 3

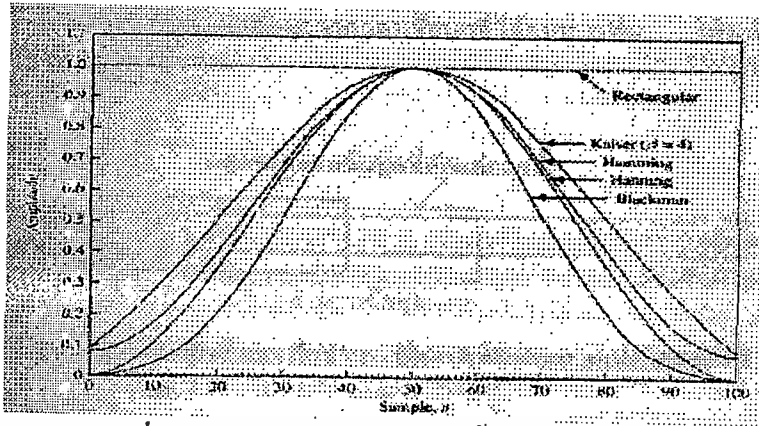
การแปลงข้อมูลเสียงให้อยู่ในรูปของสเปกโตรแกรม

สัญญาณที่ขึ้นอยู่กับเวลา นำมาแปลงให้อยู่ในรูปของสเปกโตรแกรม ซึ่งเป็นข้อมูลสามมิติ ในแกนของ เวลา-ความถี่-ความเข้ม โดยการตัดข้อมูลมาเป็นส่วนๆ เพื่อนำมาผ่านการแปลงข้อมูลให้อยู่ในแกนความถี่ การทำงานจะเลือกจากขอบเขตของสัญญาณที่จะนำมาคำนวณ ขอบเขตนี้จะเลื่อนไปในการคำนวณแต่ละครั้ง ดังสมการที่ 3.1

$$w(n, m) = \{V(nT), V((n+1)T), V((n+2)T), \dots, V((n+m)T)\} \dots \dots 3.1$$

ขอบเขตที่เลือกนี้ มีพารามิเตอร์อยู่สองตัวด้วยกันคือ n เป็นจุดเริ่มต้นของลำดับสัญญาณที่ถูกสุ่มเข้ามา พารามิเตอร์ที่สองคือ m เป็นจำนวนข้อมูลของสัญญาณที่ถูกสุ่ม ในการวิจัยนี้ได้เลือกค่า m หรือขนาดของขอบเขตเท่ากับ 256

ข้อมูลที่ถูกจำกัดด้วยขอบเขตนี้ จะนำมาผ่านสมการถ่วงน้ำหนัก ซึ่งในงานวิจัยนี้ได้ใช้วิธีการของ Welch[2] เหตุผลในการถ่วงน้ำหนักก็เพราะว่า การแปลงแบบฟูเรียร์ จะใช้กับฟังก์ชันที่มีลักษณะเป็นคาบเวลาเท่านั้น แต่สัญญาณเสียงไม่ใช่สัญญาณที่เป็นคาบเวลาอย่างแท้จริง ถ้าพิจารณาเสียงในช่วงเวลาแคบๆ จะพบว่าสัญญาณเสียงสามารถประมาณให้เป็นสัญญาณที่มีคาบได้ และเพื่อให้การประมาณนี้ถูกต้องมากยิ่งขึ้น จึงถ่วงน้ำหนักของข้อมูลที่จะนำไปใช้ในการแปลงฟูเรียร์อย่างรวดเร็ว สมการถ่วงน้ำหนัก(Windows) นี้มีอยู่หลายชนิดด้วยกัน เช่น สมการแบบ Rectangular Kaiser Hamming Hanning หรือ Blackman เป็นต้น [2],[3] กราฟของฟังก์ชันถ่วงน้ำหนักเหล่านี้มีลักษณะดังรูปที่ 3.1



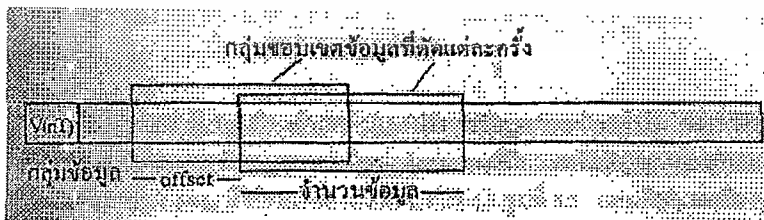
รูปที่ 3.1 กราฟแสดงฟังก์ชันถ่วงน้ำหนักชนิดต่างๆ

กลุ่มข้อมูลที่ถูกล่วงนำหนัก จะนำไปใช้เป็นข้อมูลสำหรับแปลงฟูเรียร์อย่างรวดเร็ว ข้อมูลที่ได้จะเป็นกลุ่มข้อมูลที่เป็นค่าของสัญญาณ ในแต่ละค่าความถี่ในรูปของจำนวนเชิงซ้อน ซึ่งจะต้องนำไปหาค่าพลังงานของแต่ละความถี่ ค่าพลังงานในแต่ละความถี่ที่ได้จากข้อมูลหลายๆ กลุ่มที่อยู่ต่อเนื่องกันไปในแกนของเวลานี้ เรียกสเปกโตรแกรม ซึ่งอาจจะเขียนเป็นสมการ ได้ดังนี้

$$S(T) = \{P(F(W(n,m))), P(F(W(n+off,m))), \dots, P(F(W(n+(T-1)off,m)))\} \dots 3.2$$

- โดยที่ S(T) คือสเปกโตรแกรมที่มีจำนวนชุดข้อมูล T ชุด
- W(n,m) คือกลุ่มข้อมูลที่ได้จากการถ่วงน้ำหนัก
- F(s) แทนฟังก์ชันสำหรับหาค่าฟูเรียร์อย่างรวดเร็ว
- P(s) แทนฟังก์ชันสำหรับหาค่าพลังงานของกลุ่มข้อมูลที่ได้จากการแปลงฟูเรียร์อย่างรวดเร็ว
- Off คือช่วงห่างของตำแหน่งที่ใช้ในการเลือกจุดเริ่มต้น

เราอาจแสดงการตัดข้อมูลจากกลุ่มข้อมูลที่ได้จากการสุ่ม เพื่อนำมาคำนวณหาสเปกโตรแกรม ดังรูปที่ 3.2



รูปที่ 3.2 การตัดข้อมูลเพื่อนำมาใช้ในการคำนวณหาสเปกโตรแกรม

บทที่ 4

ทฤษฎีการแปลงข้อมูลแบบคาร์ฮูเนนเลิฟ

4.1 คณิตศาสตร์การแปลงข้อมูลแบบคาร์ฮูเนนเลิฟ (Karhunen-Loeve Transform)

ในการแซมปลิง (Sampling) ข้อมูล คือ การจับสัญญาณเสียง หรือข้อมูลที่ต้องการเป็นช่วงๆ ตามความถี่การแซมปลิง ซึ่งจะเห็นได้ว่าในขั้นตอนนี้จะเป็นการทำงานแบบสุ่ม (Random Process) และข้อมูลนี้ได้จากการแซมปลิงจะแทนสัญญาณที่ต้องการ และจะนำข้อมูลที่ได้จากการสุ่มนี้มาใช้ในการประมวลผล โดยในแบบจำลองนี้จะแทนการสุ่มโดยการแซมปลิงด้วยฟังก์ชัน $x(t)$ ซึ่งเป็นฟังก์ชันในโดเมนของเวลา และสามารถเขียนเป็นสมการแทนฟังก์ชัน $x(t)$ ด้วยสมการเชิงเส้นดังสมการที่ 4.1

$$x(t) = \sum_{i=1}^N y_i \varphi_i(t) \quad 4.1$$

จากสมการที่ 4.1 ฟังก์ชัน $\varphi_i(t)$ เป็น Deterministic Function และค่าสัมประสิทธิ์ y_i จะเป็นค่าตัวแปรแบบสุ่ม (Random Variable) ซึ่งในสมการที่ 4.1 นี้จะสามารถอธิบายข้อมูล หรือสัญญาณจริง (Original Data) โดยการแปลงแบบคาร์ฮูเนนเลิฟ (Karhunen-Loeve) ได้ ซึ่งข้อมูลจริงนั้นจะประกอบด้วย ผลรวมแบบเชิงเส้นของฟังก์ชันพื้นฐาน และสัมประสิทธิ์จากการแปลง (Coefficient In The Transform Domain) ข้อมูลที่ได้จากการแซมปลิง จะเป็นข้อมูลที่มีลักษณะเป็นค่าตามเวลาแบบไม่ต่อเนื่อง (Discrete Time Domain) n ค่า ซึ่งสามารถนำค่าข้อมูลนี้มาทำให้อยู่ในรูปของเมตริกซ์ได้ดังสมการที่ 4.2

$$X = [x(t_1) \dots x(t_n)]^T \quad 4.2$$

ซึ่งเมตริกซ์นี้เรียกอีกอย่างหนึ่งว่า เวกเตอร์ ซึ่งค่า $x(t)$ จะเป็นค่าที่สุ่มออกมาตามความถี่ของการแซมปลิง และในทำนองเดียวกันสามารถสร้างค่าเวกเตอร์อีกค่าได้ ตามสมการที่ 4.3

$$\Phi = [\varphi_1(t_1) \dots \varphi_1(t_n)]^T \quad 4.3$$

และค่าข้อมูลที่ได้จากการแซมปลิง นี้จะต้องมีการทำให้ค่าเฉลี่ยของข้อมูลเป็นศูนย์ (Zero-Mean) และทำเมตริกซ์ที่ได้มานั้นเป็นเมตริกซ์จัตุรัส ($n \times n$) หรือ โคแวนเรียนซ์เมตริกซ์ (Covariance Matrix) ซึ่งสามารถทำเป็นเมตริกซ์จัตุรัสได้ตามสมการที่ 4.4

$$C = E \{ (x - \bar{m})(x - \bar{m}) \}^T \quad 4.4$$

โดย \bar{m} เป็นเวกเตอร์เฉลี่ยซึ่งคำนวณได้จากสมการที่ 4.5

$$\bar{m} = \frac{1}{M} \sum_{i=1}^M X_i \quad 4.5$$

เมื่อ M เป็นจำนวนเวกเตอร์ที่ (Dimension of Matrix) หลังจากการทำให้ค่าเฉลี่ยของข้อมูลเป็นศูนย์ (Zero-Mean) และทำให้เป็นเมตริกซ์จตุรัส (Covariance Matrix) แล้ว จะได้เมตริกซ์ที่มีขนาดแถวและหลักเท่ากัน การทำเช่นนี้เท่ากับว่าเป็นการทำให้เมตริกซ์นี้เป็นออโต้โครีเรชันเมตริกซ์ (Autocorrelation Matrix) ซึ่งมีสมการ 4.6

$$R(t, \tau) = E [x(t)x^*(\tau)] \quad 4.6$$

ถ้า $\varphi_i(t_k)$ เป็นฟังก์ชันเจาะจง (Eigen function) ซึ่งจะหาได้จาก ออโต้โครีเรชันเมตริกซ์ $R(t_p, t_k)$ เป็นไปตามสมการที่ 4.7

$$\sum_{k=1}^N R(t_p, t_k) \varphi_i(t_k) = \lambda_i \varphi_i(t_p) \quad 4.7$$

ซึ่ง $i, l = 1, 2, \dots, n$ ซึ่งสมการที่ 4.7 นี้สามารถนำมาเขียนใหม่ให้อยู่ในรูปของเมตริกซ์ได้ ซึ่ง ค่าเจาะจง (Eigenvalues) λ_i และเวกเตอร์เจาะจง (Eigenvector) Φ จะแสดงให้เห็นในซึ่ง $i, l = 1, 2, \dots, n$ ในสมการที่ 4.7 [4] นี้สามารถนำมาเขียนใหม่ในรูปเมตริกซ์ได้ตามสมการที่ 4.8 เพื่อจะแสดงค่าเจาะจง λ_i และเวกเตอร์เจาะจง Φ

$$S\Phi = \lambda_i \Phi_i \quad 4.8$$

ซึ่ง $i = 1, 2, \dots, n$ และ S เป็น $n \times n$ เมตริกซ์ หรือ เมตริกซ์จตุรัส ซึ่งสามารถแทนได้ตามสมการที่ 4.9

$$S = \begin{bmatrix} R(t_1, t_1) \dots R(t_1, t_n) \\ \vdots \\ R(t_n, t_1) \dots R(t_n, t_n) \end{bmatrix} \quad 4.9$$

และจะใช้เมตริกซ์จัตุรัสนี้ในการหาค่าเจาะจง (Eigen values) เพราะทุกๆ เมตริกซ์จัตุรัสจะสามารถหาค่าเจาะจงได้ และเวกเตอร์เจาะจงซึ่งสัมพันธ์กันกับค่าเจาะจง ซึ่งเวกเตอร์เจาะจง (Eigen vector) จะมีคุณสมบัติที่เรียกว่า ออร์โธโนมัล (Orthonormal) ดังสมการที่ 4.10

$$\Phi^T \Phi = I \quad 4.10$$

โดย I เป็นเมตริกซ์เอกลักษณ์ (Identity Matrix)

ค่าเจาะจง (Eigen values) จะถูกจัดเรียงตามลำดับความสำคัญโดยจะเทียบกับผลรวมของพลังงานทั้งหมด ซึ่งก็คือผลรวมกับค่าเจาะจง และค่าเจาะจงที่ได้จะถูกเลือกตามลำดับจากมากไปน้อย ($\lambda_1 \geq \lambda_2 \geq \dots \lambda_n$) และค่าเจาะจงจะสามารถนำไปใช้หาค่าเวกเตอร์เจาะจง (Eigen vector) ที่มีความสัมพันธ์กันได้โดยค่าเจาะจง n ตัว จะได้ค่าเวกเตอร์เจาะจง n ชุด ซึ่งจะกล่าวถึงวิธีหาค่าต่างๆ จากเมตริกซ์จัตุรัสต่อไป

เวกเตอร์เจาะจงที่ได้จากค่าเจาะจงจะแทนฟังก์ชันมูลฐานในการแปลงโดยวิธีคาร์ฮูเนน-โลฟ (Karhunen-Loeve) ซึ่งจะอยู่ในรูปโดเมนที่ถูกแปลงแล้ว (Transformed Domain) ซึ่งเรียกว่า KL Space และค่าสัมประสิทธิ์ของ KL (KL Coefficient) y_i จะคำนวณได้โดยการถ่ายถอดข้อมูลจริงไปบนโดเมนใหม่ ซึ่งแทนได้โดยฟังก์ชันพื้นฐาน (Projecting) ตามสมการที่ 4.11

$$y_i = \Phi^T X_i \quad 4.11$$

ซึ่งจะทำให้ได้ค่าเวกเตอร์สัมประสิทธิ์ (Coefficient Vector) y_i หนึ่งตัวมีความสัมพันธ์กันกับเวกเตอร์เจาะจง (Eigen vector) หนึ่งชุดซึ่งจะทำให้ได้ สมการที่ 4.12

$$Y = [y_1 \dots y_n]^T \quad 4.12$$

เป็นค่าเวกเตอร์ที่ประกอบด้วยค่าสัมประสิทธิ์ n ตัว ซึ่งสามารถคำนวณได้จากเวกเตอร์เจาะจง

(Eigen vector) $\Phi = [\varphi_1 \dots \varphi_n]^T$ ซึ่ง Y จะเป็นการแปลงแบบออร์โธโนมัล (Orthonormal Transformation) จากเวกเตอร์ X ที่มาจากการสุ่มค่าโดยการแซมปลิง (Sampling) ไปเป็นเวกเตอร์ใหม่ KL Space ซึ่งก็เป็นค่าเวกเตอร์แบบสุ่มด้วย ในการสร้างข้อมูลเดิมขึ้นมาใหม่ โดยใช้เวกเตอร์เจาะจง และค่าเวกเตอร์สัมประสิทธิ์ (Coefficient Vector) ที่มีความสัมพันธ์กัน จากค่าทั้งสองสามารถสร้างข้อมูลจริง (Original Data) จากสมการที่ 4.13

$$X = \Phi Y \quad 4.13$$

4.2 คำอธิบายเสริมแบบจำลองทางคณิตศาสตร์ [4]

4.2.1 การหาค่าทรานสโพสของเมทริกซ์ (Transpose of Matrix)

นิยามของการทรานสโพส (Transpose) ถ้า $A = [a_{ij}]_{m \times n}$ แล้ว ทรานสโพสของ A คือ (A^T) $A^T = [a_{ji}]_{n \times m}$ ตัวอย่างการทรานสโพส

$$\text{ถ้า } A = \begin{bmatrix} 1 & 3 \\ 2 & 4 \\ 6 & 5 \end{bmatrix}_{3 \times 2} \quad A^T = \begin{bmatrix} 1 & 2 & 6 \\ 3 & 4 & 5 \end{bmatrix}_{2 \times 3}$$

4.2.2 การทำเมทริกซ์จัตุรัส ($n \times n$) หรือ โคเวเรียนซ์เมทริกซ์ (Covariance Matrix)

โคเวเรียนซ์เมทริกซ์ (Covariance Matrix) หรือ cov หาได้จาก $\text{cov } A = A \cdot A^T$ เมื่อ A แทนเมทริกซ์ขนาด $m \times n$ ตัวอย่างการหาโคเวเรียนซ์เมทริกซ์

$$\text{ถ้า } A = \begin{bmatrix} 1 & 3 \\ 2 & 4 \\ 6 & 5 \end{bmatrix}_{3 \times 2} \quad \text{cov } A = \begin{bmatrix} 1 & 3 \\ 2 & 4 \\ 6 & 5 \end{bmatrix}_{3 \times 2} \cdot \begin{bmatrix} 1 & 2 & 6 \\ 3 & 4 & 5 \end{bmatrix}_{2 \times 3} = \begin{bmatrix} 10 & 14 & 21 \\ 14 & 20 & 32 \\ 21 & 32 & 61 \end{bmatrix}_{3 \times 3}$$

4.2.3 การหาค่าเจาะจง (Eigenvalue) และเวกเตอร์เจาะจง (Eigenvector)

นิยาม ถ้า A เป็นเมทริกซ์ขนาด $n \times n$ แล้วเวกเตอร์ X ที่ไม่เป็นเวกเตอร์ศูนย์ จะเรียกว่า เวกเตอร์เจาะจง (Eigenvector) ของ A ถ้า AX เป็นผลคูณของสเกลาร์กับ X กล่าวคือ $AX = \lambda X$ สำหรับบางสเกลาร์ λ ว่า ค่าเจาะจง (Eigenvalue) ของ A และเรียกเวกเตอร์ X ว่า เป็นเวกเตอร์เจาะจงที่สมนัยกับ λ

ตัวอย่าง เวกเตอร์ $X = \begin{bmatrix} 1 \\ 2 \end{bmatrix}_{2 \times 1}$ เป็นเวกเตอร์เฉพาะของ $A = \begin{bmatrix} 3 & 0 \\ 8 & -1 \end{bmatrix}_{2 \times 2}$ ที่สมนัยกับค่า

$$\text{เฉพาะ } \lambda = 3 \text{ เมื่อ } AX = \begin{bmatrix} 3 & 0 \\ 8 & -1 \end{bmatrix}_{2 \times 2} \cdot \begin{bmatrix} 1 \\ 2 \end{bmatrix}_{2 \times 1} = \begin{bmatrix} 3 \\ 6 \end{bmatrix}_{2 \times 1} = 3X$$

วิธีการหาค่าเฉพาะ และเวกเตอร์เฉพาะของเมทริกซ์ขนาด $n \times n$

$$\text{จากนิยาม } AX = \lambda X$$

$$\text{เขียนใหม่ได้ } AX = \lambda IX$$

$$\text{หรือ } (A - \lambda I)X = 0 \quad 4.14$$

สำหรับ λ ที่เป็นค่าเฉพาะ ต้องมีผลเฉลยที่ไม่เป็นศูนย์ของระบบสมการนี้ จากสมการ 4.14 เราจะมีผลเฉลยที่ไม่เป็นศูนย์ได้ก็ต่อเมื่อ

$$\det(A - \lambda I) = 0 \text{ หรือ } |A - \lambda I| = 0 \quad 4.15$$

เรียกสมการ 4.15 ว่าเป็น สมการลักษณะเฉพาะ (Characteristic Equation) ของเมทริกซ์ A สเกลาร์ λ ที่สอดคล้องกับสมการนี้คือ ค่าเฉพาะของ A เมื่อกระจายตัวกำหนด $\det(A - \lambda I)$ จะเป็นพหุนามใน λ เราเรียกว่า พหุนามลักษณะเฉพาะ (Characteristic Polynomial) ของ A

ตัวอย่างการหาค่าเฉพาะ และ เวกเตอร์เฉพาะ ถ้า $A = \begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix}_{2 \times 2}$

สมการลักษณะเฉพาะของ A คือ

$$|A - \lambda I| = 0$$

$$\begin{vmatrix} 1-\lambda & 2 \\ 2 & 1-\lambda \end{vmatrix} = 0$$

$$\lambda^2 - 2\lambda - 3 = 0$$

$$(\lambda+1)(\lambda-3) = 0$$

ดังนั้นค่าเฉพาะของ A คือ $\lambda = -1$ และ $\lambda = 3$

หาเวกเตอร์เฉพาะได้จาก $(A - \lambda I)X = 0$

สำหรับ $\lambda = -1$ จะได้

$$\begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}_{2 \times 2} \cdot \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}_{2 \times 1} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}_{2 \times 1} \text{ เมื่อ } \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}_{2 \times 1} \text{ เป็นเวกเตอร์เฉพาะ}$$

$$2X_1 + 2X_2 = 0$$

$$X_1 = -X_2$$

เพราะฉะนั้นเวกเตอร์เฉพาะที่สมนัยกับ $\lambda = -1$ คือ $k \begin{bmatrix} -1 \\ 1 \end{bmatrix}_{2 \times 1}$ เมื่อ k เป็นค่าคงที่

ใดๆ ที่ไม่เป็นศูนย์ สำหรับ $\lambda = 3$ จะได้

$$\begin{bmatrix} -2 & 2 \\ 2 & -2 \end{bmatrix}_{2 \times 2} \cdot \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}_{2 \times 1} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}_{2 \times 1}$$

$$-2X_1 + 2X_2 = 0$$

$$2X_1 - 2X_2 = 0$$

$$X_1 = X_2$$

ดังนั้นเวกเตอร์เฉพาะที่สมนัยกับ $\lambda = 3$ คือ $m \begin{bmatrix} 1 \\ 1 \end{bmatrix}_{2 \times 1}$ เมื่อ m เป็นค่าคงที่ใดๆ ที่ไม่เป็น

ศูนย์

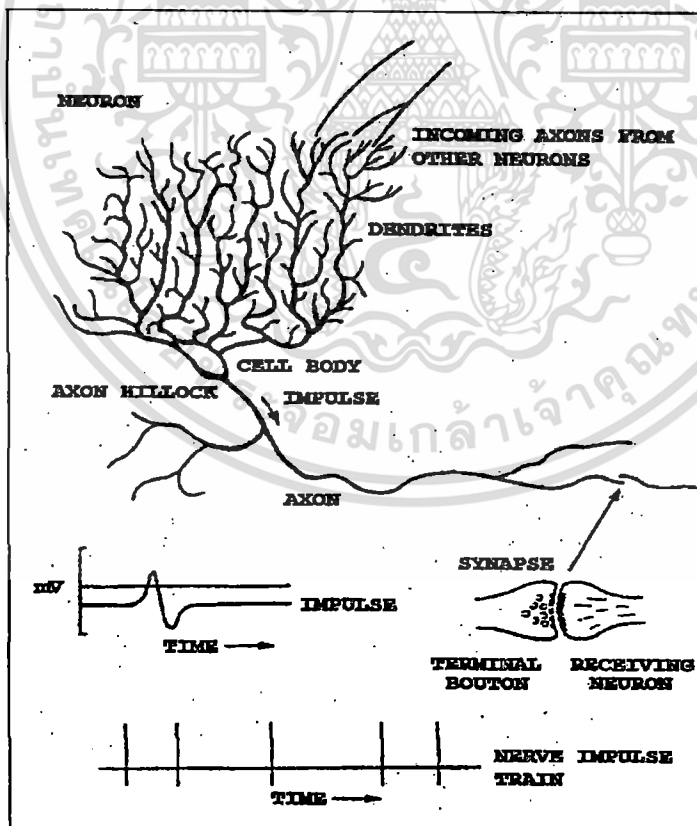


บทที่ 5

ความรู้เบื้องต้นเกี่ยวกับโครงข่ายประสาทเทียม

ในช่วงระยะเวลา 8-9 ปีที่ผ่านมา ในต่างประเทศมีการตื่นตัวในการวิจัยและพัฒนาเกี่ยวกับโครงข่ายประสาทเทียม (Artificial Neural Networks: ANN) อย่างกว้างขวาง ทั้งทางด้านทฤษฎีและการประยุกต์ใช้งาน จนกระทั่งถึงปัจจุบันได้มีการประยุกต์นำโครงข่ายประสาทเทียม มาใช้ในอุปกรณ์เครื่องใช้ต่างๆ มากขึ้น การวิจัยสร้างโครงข่ายประสาทเทียม (Artificial Neural Networks) มีแนวคิดเลียนแบบการทำงานของสมองชีวภาพ โดยเรียนรู้และศึกษาการทำงานของสมองชีวภาพเพื่อกำหนดแนวทางสำหรับการสร้างแบบจำลองขึ้นมา แล้วพยายามสมมติฐานลักษณะการทำงานโดยจำลองเป็นโมเดลคณิตศาสตร์ที่มีลักษณะเดียวกันแล้วดำเนินการคำนวณโดยใช้คอมพิวเตอร์

ระบบคิดคำนึงของมนุษย์ มีโครงสร้างพื้นฐานจากเซลล์สมองที่เรียกว่านิวรอน (Neurons) เรียงเป็นชั้นๆ อย่างซับซ้อนจำนวนมหาศาล ประมาณหมื่นล้าน (10^{11}) นิวรอนและอาจมีจุดเชื่อมโยงส่งผ่านจุดเชื่อมโยงภายในถึงพันล้านล้าน (10^{15}) จุด (Philip D. Wasserman, 1989:12) แต่ละนิวรอนจะมีคุณลักษณะแตกต่างกันไปโดยมีการทำงานคล้ายกันคือ รับเข้า, ประมวลผล, ส่งออก สัญญาณไฟฟ้าเคมีผ่านไปยังนิวรอน ซึ่งจะส่งสื่อสารไปตามระบบของสมอง

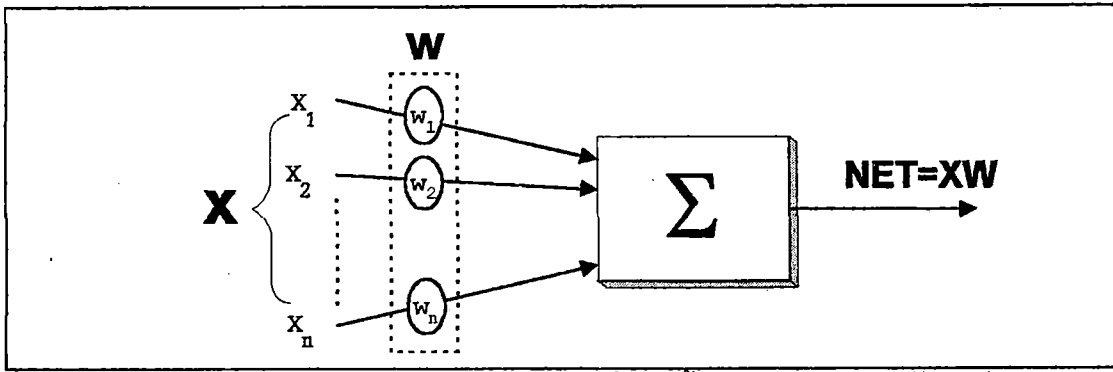


รูปที่ 5.1 แสดงโครงสร้างตัวอย่างของเซลล์ประสาทชีวภาพ

จากรูปที่ 5.1 ส่วนแขนที่ขยายแยกออกไปจากตัวเซลล์ต่อไปยังเซลล์อื่น ๆ เพื่อรับสัญญาณเรียกว่า เดนไดรท์ (Dendrites) จุดรับสัญญาณจากเซลล์อื่นเข้ามายังตัวเซลล์จะผ่านมาทางจุดเชื่อมต่อที่เรียกว่า ซินแนปส์ (Synapse) ซึ่งแอกซอน (Axon) จะเป็นตัวส่งสัญญาณเอาท์พุทออกไปยังนิวรอลอื่นจากผลการวิจัยพบว่า แต่ละนิวรอลจะเชื่อมต่อออกไปยังนิวรอลอื่นๆ ซึ่งแต่ละนิวรอลจะมีคุณสมบัติในการเพิ่มขยายหรือลดทอนความเข้มของสัญญาณบางสัญญาณที่เข้ามาทางเดนไดรท์ ของเซลล์ (ซึ่งมีแขนมากมาย) อาจสามารถกระตุ้นตัวเซลล์ แต่บางสัญญาณก็อาจจะยับยั้งตัวเซลล์ เนื่องจากเซลล์ประสาทหนึ่งเซลล์มีเดนไดรท์มาก ฉะนั้น สัญญาณ กระตุ้นจากเดนไดรท์ ที่รับเข้ามาจากเซลล์ประสาทอื่นๆ จะถูกนำมารวมกันที่ตัวเซลล์ประสาทที่เซลล์ประสาทจะมีค่าเทรชโฮลด์ (Threshold) ค่าหนึ่งหากผลรวมของสัญญาณไฟฟ้าเคมี (Electrochemical) มีค่ามากกว่า เทรชโฮลด์เซลล์ เซลล์ประสาทก็จะส่งสัญญาณขนาดหนึ่งผ่านทางแอกซอนไปยังนิวรอลอื่นๆ การจัดเรียงชั้น (Layer) และลักษณะการเชื่อมโยงระหว่างนิวรอลในสมองนั้นมีการจัดเรียงที่ซับซ้อน สอดคล้องกับหน้าที่การทำงานเฉพาะส่วน มีการเจริญเติบโตสัมพันธ์กับสิ่งแวดล้อมและมีการเรียนรู้ตลอดเวลาซึ่งใช้เวลานานนับปีดังนั้นจึงยากที่จะสร้างโมเดลขึ้นมาเพื่อเลียนแบบให้มีคุณลักษณะคล้ายสมองชีวภาพได้ทั้งหมด ผลงานที่ได้จากการทำวิจัยในปัจจุบันเป็นเพียงการจำลอง การเลียนแบบและการทำงานเฉพาะบางส่วนของโครงข่ายประสาทมาใช้เฉพาะกับงานใดงานหนึ่ง ซึ่งมีการวิจัยลักษณะของโครงข่ายแบบต่างๆ ขึ้นมา โดยแต่ละแบบจะเหมาะกับงานประเภทหนึ่ง ๆ เท่านั้น

5.1 โครงข่ายประสาทเทียม (Artificial Neural Networks)

การออกแบบสร้างประสาทเทียมนั้นมีสมมติฐานขั้นแรกจากคุณสมบัติของระบบประสาทที่ชีวภาพ ดังที่กล่าวมา กล่าวคือ ชุดรับสัญญาณข้อมูล อินพุทของเซลล์ประสาทหนึ่งได้จากสัญญาณเอาท์พุทของเซลล์ประสาทอื่นๆ ผ่านทางซินแนปส์และเดนไดรท์ ข้อมูลแต่ละค่าที่รับมาจะถูกลดขนาดด้วย ซินแนปติกส์ ซึ่งภายในประกอบด้วยสารเคมีประเภท K^+ , Ca^{++} , Na^+ , Cl^- ซึ่งจะมีลักษณะทางความนำ พัลส์ (Pulse) สัญญาณไฟฟ้าเคมีที่แตกต่างกัน (James A. Freeman and David M. Skapura, 1991:8-9) ด้วยเหตุนี้ โมเดลประสาทเทียมที่สร้างขึ้น จะต้องมีการถ่วงน้ำหนักให้กับโมเดลก่อนที่จะนำเข้าสู่โมเดลประสาทเทียม จุดนี้เรียกว่า ซินแนปติกส์เวกท์ ปริมาณของข้อมูลที่เข้าสู่นิวรอล จะถูกนำมารวมกัน และตัดสินใจด้วยระดับความสนใจของ นิวรอล (Activation level) แล้วจะส่งเป็นเอาท์พุทออกที่แอกซอนไปยังนิวรอลอื่นๆ



รูปที่ 5.2 แสดงไดอะแกรมของนิวรอนที่สร้างขึ้น (Artificial Neuron)

จากรูปที่ 5.2 แสดงถึงโมเดลที่สร้างขึ้นโดยแนวความคิดจากเซลล์สมองชีวภาพ สัญญาณอินพุต คือ X_1, X_2, \dots, X_n จะถูกป้อนเข้าไปยังนิวรอนที่สร้างขึ้น ซึ่งเปรียบเทียบกับสัญญาณที่ป้อนเข้ายังซินแนปส์ของ นิวรอนชีวภาพ สัญญาณอินพุตนี้จะนำไปคูณกับค่าซินแนปติกส์เวกท์ที่มีค่าตั้งแต่ 0.00-1 (Weight: ค่าที่ใช้ถ่วงน้ำหนัก) W_1, W_2, \dots, W_n ก่อนที่จะเข้าสู่บล็อกซัมเมชัน (Σ : Summation) ซึ่งค่าถ่วงน้ำหนักนี้จะสอดคล้องกับค่าสเตรงท์ (Strength) ของจุดต่อซินแนปส์ชีวภาพแต่ละจุด (Single biological synaptic connection) บล็อกซัมเมชันนี้ก็จะทำหน้าที่สอดคล้องคล้ายกับตัวเซลล์สมองชีวภาพ ผลรวมทางคณิตศาสตร์ของอินพุตและเวกท์จะได้เป็นเอาท์พุท เราเรียกว่า เน็ต (NET) ซึ่งเราจะรวมกันในรูปของเวกเตอร์ได้ดังนี้

$$NET = X_1W_1 + X_2W_2 + \dots + X_nW_n \dots\dots\dots(5.1)$$

จะได้

$$NET = XW \dots\dots\dots(5.2)$$

5.2 ฟังก์ชันกระตุ้นความสนใจ (Activation Function)

เมื่อได้สัญญาณ NET แล้วกระบวนการต่อมาที่นิวรอนต้องทำคือตัดสินใจเราจึงต้องกำหนดฟังก์ชันการ ตัดสินใจ เพื่อใช้เป็นระดับของการตัดสินใจให้กับนิวรอน เพื่อให้ได้สัญญาณเอาท์พุทของ นิวรอนออกมา ซึ่งเชื่อมต่อไปยังนิวรอนตัวอื่น ๆ เป็นโครงข่าย OUT ที่ได้อาจเป็น Simple linear function โดย

$$OUT = K[NET] \dots\dots\dots(5.3)$$

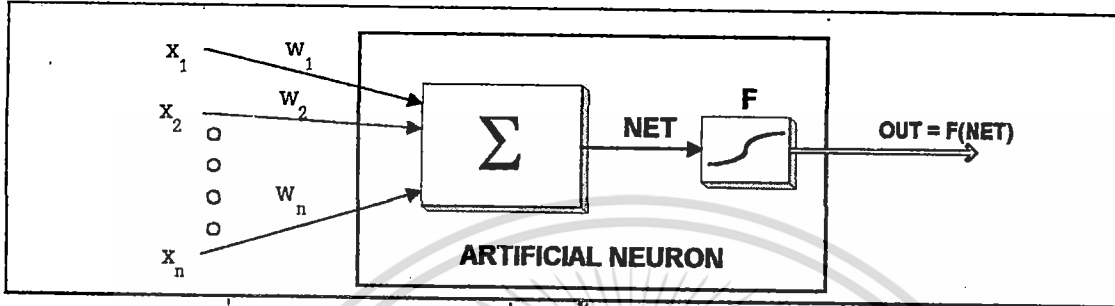
โดย K เป็นค่าคงที่ ที่เรียกว่า Threshold function

ตัวอย่างเช่น

OUT = 1 ถ้า NET > T

OUT = 0 เมื่อเป็นกรณีอื่น(5.4)

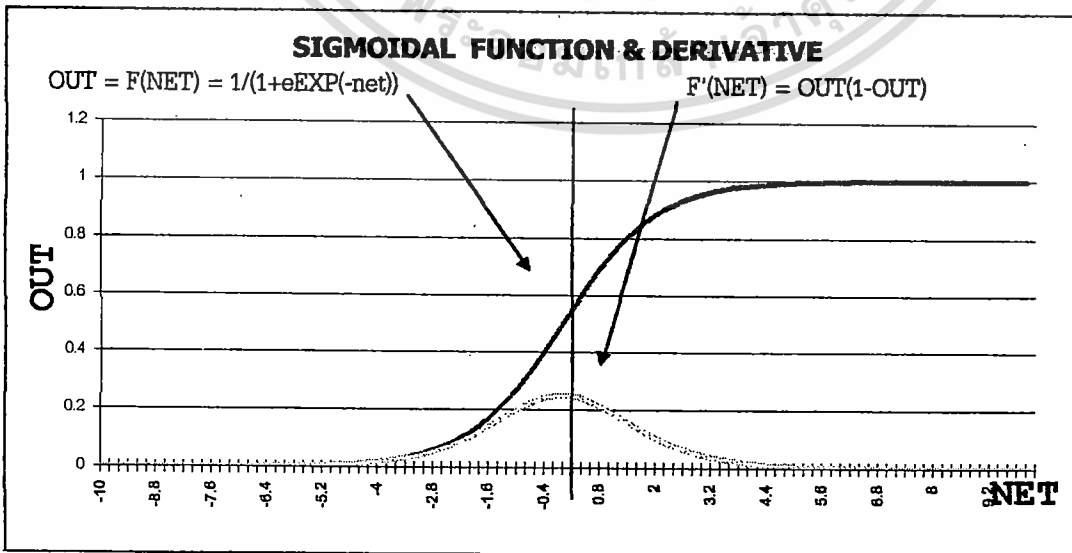
และ T เป็นค่าเทรชโฮลคั้งที่ หรืออาจเป็น Function อื่น ๆ ที่เลียนแบบคุณสมบัติที่ไม่เป็นเชิงเส้นของเซลล์ประสาทซึ่งภาพได้อย่างใกล้เคียงกว่า และใช้เป็นฟังก์ชันให้กับโครงข่ายทั่วไปได้



รูปที่ 5.3 แสดงโมเดลนิวรอนที่สร้างขึ้นร่วมกับ Activation Function

ในรูปที่ 5.3 บล็อก F จะรับผลที่ได้จาก NET มาสร้างเป็นสัญญาณเอาต์พุตที่ OUT โดยกระบวนการภายในบล็อก F จะมีช่วงของ OUT ให้อยู่ในขอบเขตจำกัด ตามต้องการ ดังนั้น ค่า OUT จะมีค่าไม่ต่ำกว่าช่วงที่กำหนดโดยค่าของ NET เราเรียกบล็อก F นี้ว่าสควาซิงฟังก์ชัน (Squashing function) และโดยทั่วไป สควาซิงฟังก์ชันที่ใช้เป็นแบบลอจิสติกฟังก์ชัน หรือซิกมอยด์ (Logistic function or "Sigmoid") ซึ่งมีรูปร่างคล้ายตัว S โดยเขียนเป็นสมการคณิตศาสตร์ได้ดังนี้คือ

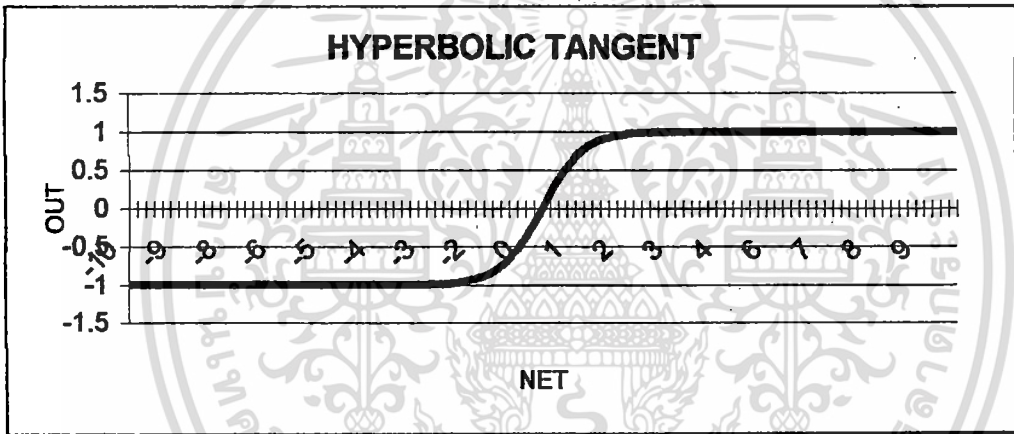
$$F(x) = \frac{1}{(1 + e^{-x})} \dots\dots\dots(5.5)$$



รูปที่ 5.4 แสดงกราฟที่ได้จากสมการซิกมอยด์ลอจิสติกฟังก์ชัน (Sigmoidal logistic function)

ลักษณะของเทรซไฮลฟังก์ชันมีลักษณะเป็น Non-linear function เช่น S-Curve เราจะได้ค่าเอาต์พุต ที่มีความไวต่อสัญญาณอินพุตที่มีขนาดเล็กๆ และเฉื่อยต่อสัญญาณแรงๆ ซึ่งสัญญาณอ่อนๆ ไปทางบวกเพียงเล็กน้อยก็จะให้ OUT ใกล้เคียง "1" กระตุ้นหรือสัญญาณอ่อนๆ ทางลบเพียงเล็กน้อยก็จะทำให้ Output ใกล้เคียง "0" (ยับยั้ง) ขณะที่สัญญาณแรงๆ ทางบวกก็ยังคงให้ Output ใกล้เคียง "1" และสัญญาณทางลบแรงๆ ก็คงให้ Output ใกล้เคียง "0" เช่นกัน คุณลักษณะแบบนี้ เป็นแบบ NON-LINEAR GAIN ซึ่งคลออสส์เบอร์ก (Grossberg,1973) พบว่า คุณลักษณะที่เป็น Non-linear gain นี้สามารถแก้ปัญหา Noise-saturation dilemma ได้ และทำให้นิวรอลเทียมที่สร้างขึ้นสามารถทำงานกับขนาดของอินพุตได้กว้างมากขึ้น

ยังมีฟังก์ชันอื่นๆ อีกคือ ไฮเพอร์โบลิก แทนเจนท์ (Hyperbolic tangent) มันจะมีลักษณะคล้ายกับ Logistic function และนิยมใช้บ่อยๆ ในการสร้างโมเดลคณิตศาสตร์ การกระตุ้นเร้าความสนใจของเซลล์สมองเทียม ซึ่งมีคุณสมบัติคล้ายชีวภาพของเซลล์สมอง คือ $OUT = \text{Tanh}(X)$



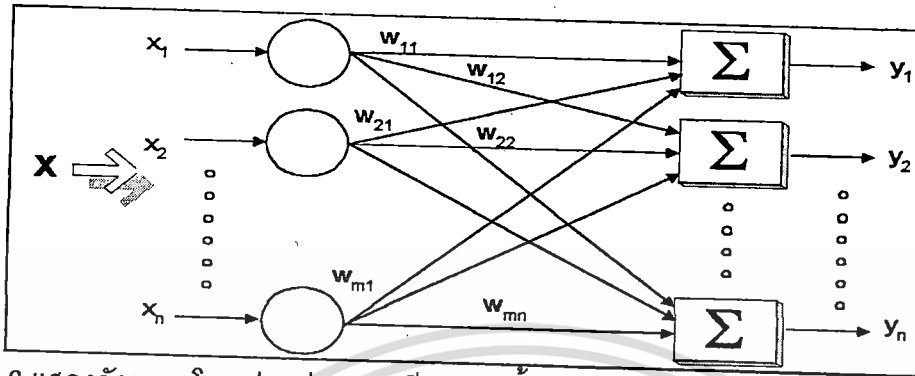
รูปที่ 5.5 แสดง Hyperbolic Tangent Function

จากรูปที่ 5.5 ส่วนที่เหมือนกับ ซิกมอยด์ ลอจิสติก ฟังก์ชัน คือมีลักษณะเป็น S แต่เนื่องจากมันจะมีความสมมาตรจึงให้ OUTPUT อยู่ระหว่าง "-1" ถึง "1" OUTPUT จะเป็น "0" เมื่อ NET เป็น "0" OUTPUT เข้าใกล้ "1" เมื่ออินพุตไปทางบวกและเข้าใกล้ "-1" เมื่อ อินพุตมีทิศทางไปทางลบ

5.3 โครงข่ายประสาทเทียมแบบชั้นเดียว (Single Layer Artificial Neural Networks)

ที่กล่าวมาจนถึงจุดนี้ เป็นการกล่าวถึงหลักการและเหตุผลในการสร้างเซลล์ประสาทเทียมเพียง หนึ่งเซลล์ โดยใช้แนวความคิดจากเซลล์ประสาทชีวภาพ การจะนำเซลล์ประสาทเทียมมาใช้งานได้นั้น ต้องใช้เซลล์ประสาทเทียมที่มีคุณลักษณะต่างๆ กัน (ค่า Weight จะทำให้คุณสมบัติของเซลล์ประสาทเทียมแต่ละเซลล์มีคุณลักษณะแตกต่างกันไป) มาเชื่อมโยงเป็นโครงข่ายในลักษณะ

เดียวกับเซลล์สมองชีวภาพเสียก่อน ซึ่งลักษณะการเชื่อมโยงมีหลายชนิด แต่ละชนิดก็มีคุณลักษณะเด่นที่แตกต่างกันไป



รูปที่ 5.6 แสดงลักษณะโครงข่ายประสาทเทียมแบบชั้นเดียว (Single-Layer Neural Networks)

จากรูปที่ 5.6 เป็นโครงข่ายประสาทเทียมแบบชั้นเดียว ที่ประกอบด้วยเซลล์ประสาทเทียมง่ายๆ หลายๆชุด ความสามารถในการคำนวณของโครงข่ายประสาทเทียมได้มาจากลักษณะการเชื่อมต่อเป็นโครงข่ายประสาทเทียมโครงข่ายง่ายๆ เป็นกลุ่มโมดูลประสาทเทียมที่เชื่อมต่อกันเป็นชั้น ๆ (Layer) ในรูปที่ 5.7 เป็นโครงข่ายประสาทเทียม แบบชั้นเดียว (Single layer) ที่ประกอบด้วยเอาต์พุทเลเยอร์ (กลุ่มของบล็อกซิมเมตริกที่อยู่ทางขวามือ) และอินพุทเลเยอร์(วงกลมทางซ้ายมือ)โดยไม่พิจารณาอินพุทเลเยอร์ว่าเป็นนิวรอลเลเยอร์ เนื่องจากอินพุทเลเยอร์จะทำหน้าที่เชื่อมต่ออินพุทที่รับมาและส่งออกไปให้ยังแต่ละอินพุทนิวรอลเลเยอร์ (ในที่นี้คือ Output layer) ในชั้นถัดไป โดยแต่ละอินพุทจะถูกคูณโดยค่าเวกท์เฉพาะแต่ละอินพุท โครงข่ายประสาทเทียมที่สร้างขึ้นในชั้นแรกไม่ซับซ้อน โดยแต่ละนิวรอล จะได้เอาต์พุทจาก

$$OUT = \text{Logistic Function} \text{ คูณ (ผลรวมของ Input X กับ Weight)}$$

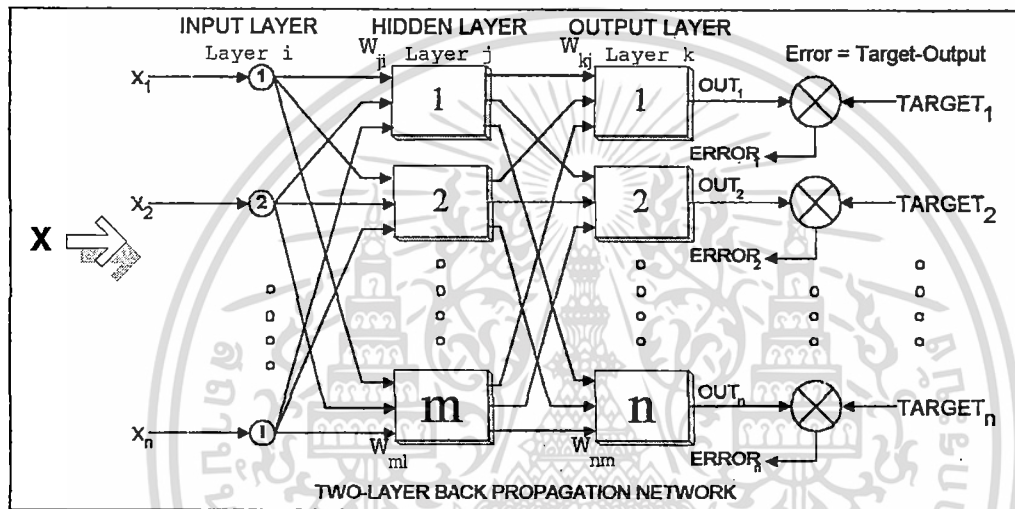
หรือ

$$OUT = F(NET) \dots\dots\dots(5.6)$$

อย่างไรก็ดีลักษณะการเชื่อมโยงระหว่างโครงข่ายไม่ได้มีแบบเดียว การเชื่อมโยงระหว่างเลเยอร์อาจมีการเชื่อมโยงย้อนกลับมาจากอินพุทเลเยอร์อีก ซึ่งโครงข่ายประสาทชีวภาพก็มีลักษณะดังกล่าวเช่นกัน สำหรับค่า Weight ในรูปที่ 5.7 มีวิธีการพิจารณาในรูปของ เวกท์เมตริก (Weight matrix) ซึ่งหากโครงข่ายมีหลายชั้น จะช่วยให้ระบุค่าเวกท์ได้ง่ายขึ้น และเพื่อหลีกเลี่ยงความสับสนจะกำหนดเป็น ไดมension (Dimensions) ของเมตริก โดยให้ m แทนจำนวนแถว หรือจำนวนของอินพุทและ n แทนจำนวนของนิวรอล ที่สร้างขึ้น ตัวอย่างเช่น เวกท์ ที่เชื่อมระหว่างอินพุท ตัวที่ 4 กับนิวรอลตัวที่ 2 คือ $w_{4,2}$

5.4 โครงข่ายประสาทเทียมแบบหลายชั้น (Multilayer Artificial Neural Networks)

โครงข่ายที่ซับซ้อนจะมีความสามารถในการคำนวณที่ดีขึ้นมันจะเป็นโครงข่ายที่มีโครงสร้างเป็นจินตนาการที่น่าเป็นไปได้โดยการจัดการเชื่อมโยงนิเวศมีโครงสร้างเป็นชั้นๆ คล้ายส่วนหนึ่งของสมอง และมีการพัฒนาอัลกอริทึมเกี่ยวกับการฝึกสอนให้โครงข่ายแบบหลายชั้นทำงานได้ตามความต้องการแล้วเมื่อไม่นานมานี้ โครงข่ายแบบหลายชั้น อาจจะสร้างจาก กลุ่มของโครงข่ายแบบชั้นเดียวเอาที่ทุกของ Layer หนึ่ง จะใช้เป็นอินพุทของ Layer ถัดไป ในภาพที่ 22 แสดงเน็ตเวิร์คที่มีการเชื่อมต่อแบบสองชั้น



รูปที่ 5.7 แสดงไดอะแกรมของ Backpropagation Neural Networks แบบสองชั้น

รูปที่ 5.7 แสดงโครงข่ายประสาทเทียมแบบหลายชั้น ที่ต่อเชื่อมโยงแบบเต็มชั้น ในโครงข่ายแบบหลายชั้นมีการเรียกชื่อชั้นต่างๆ ดังนี้ คือ ชั้นที่ต่อโดยตรงกับอินพุท เรียกว่า อินพุทเลเยอร์ (Input layer) ชั้นนี้จะไม่มีการคำนวณ แต่จะทำหน้าที่ต่อเชื่อมข้อมูลไปยังชั้นถัดไป ชั้นที่อยู่ท้ายสุดทางขวามือเรียกว่า เอาท์พุทเลเยอร์ (Output layer) เป็นชั้นที่โครงข่ายจะให้ผลลัพธ์ ส่วนชั้นที่อยู่ระหว่างอินพุทเลเยอร์และเอาท์พุทเลเยอร์ จะมีกี่ชั้นก็ตามจะเรียกว่า ฮิดเดนเลเยอร์ (Hidden layer) หากฮิดเดนเลเยอร์ มีหลาย ๆ ชั้นก็จะมีการตั้งชื่อเฉพาะลงไปให้กับแต่ละชั้น

5.5 การฝึกสอนให้กับโครงข่ายประสาทเทียม (Training of Artificial Neural Networks)

ค่าเวกซ์ มีความสัมพันธ์กับอะไร ? เปลี่ยนแปลงอย่างไร ? นั่นก็เช่นเดียวกับเด็กที่คลอดออกมาที่มีสมองแล้วแต่สมองยังไม่เจริญเติบโตเพียงพอและยังไม่ได้รับการฝึกสอนและเรียนรู้ เด็กจึงไม่สามารถทำกิจกรรมใดๆ ด้วยตนเอง เว้นแต่กิจกรรมที่ธรรมชาติสร้างมาพร้อมกับการกำเนิดที่เรียกว่า "สัญชาตญาณ" ซึ่งธรรมชาติใส่คุณลักษณะบางอย่างให้เซลล์สมองบางส่วนตั้งแต่ทารกเจริญเติบโตอยู่ในครรภ์มารดา เช่น ระบบควบคุมการหายใจ, ความรู้สึก, การเรียกร้องเมื่อหิว, การ

ตอบสนองต่อสิ่งเร้า ฯลฯ เด็กจะพัฒนาการเรียนรู้ไปตามขั้นตอน หลังจากนั้นสมองของเขาจะได้รับการฝึกสอน และเจริญเติบโตไปพร้อมๆ กัน เซลล์สมองจะได้รับการปรับคุณลักษณะสอดคล้องกับการฝึกสอน และจะเจริญเป็นโครงข่ายสอดคล้องกัน

โครงข่ายประสาทเทียมที่สร้างขึ้นมีลักษณะเช่นเดียวกัน คือ เมื่อสร้างเสร็จ แต่ละเซลล์ประสาทที่สร้างขึ้นมานั้น จะยังไม่มีคุณลักษณะใดเลย เนื่องจากยังไม่มีกำหนดค่าซินแนปติกส์เวกซ์ที่เหมาะสมกับงานที่ต้องการให้กับมัน จึงต้องมีการฝึกสอนเพื่อให้เน็ตเวิร์คที่สร้างขึ้นมีคุณลักษณะตามที่ต้องการ การฝึกสอนของโครงข่ายประสาทเทียม จะกระทำโดยการปรับเปลี่ยนค่าซินแนปติกส์เวกซ์เพื่อให้โครงข่ายจดจำ แพตเทิร์นความสัมพันธ์ระหว่างอินพุทกับเอาต์พุทได้ โดยในขั้นแรกอาจกำหนดเป็นค่าสุ่มใดๆ (Random weight) ก่อนแล้วถึงปรับเปลี่ยนเวกซ์ไปตามอัลกอริทึมที่เหมาะสม หลายๆ รอบจนกว่าจะได้เอาต์พุทของเน็ตเวิร์ค เหมือนกับเอาต์พุทที่ต้องการในเรื่องไขความผิดพลาดที่ยอมรับได้

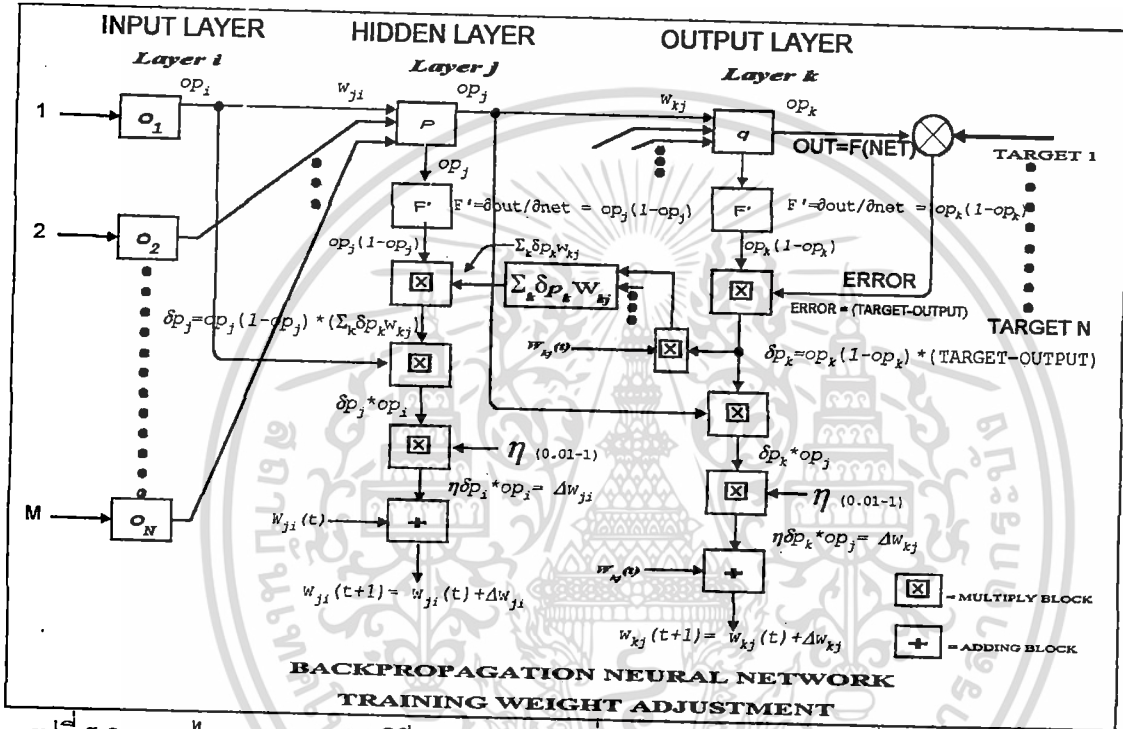
5.6 วิธีการแก้ปัญหาการฝึกสอน (Traning Algorithm)

ส่วนใหญ่แล้วทุกวันนี้ การแก้ปัญหาฝึกสอนของโครงข่ายค่อยๆ พัฒนาก้าวหน้าขึ้นจากแนวความคิดของ ดี โอ เฮบบ์ (ปี 1961) เขาได้เสนอโมเดลของ การเทรนนิ่ง แบบอิสระ (Unsupervised training) ในแบบซินแนปติกส์สเตรงท์ หรือเวกซ์ ซึ่งจะเพิ่มขึ้น ถ้าทั้งแหล่งกำเนิด (Input Source) และจุดหมายปลายทาง (Destination) ของนิวรอลได้รับการสนใจ กรณีนี้ถ้ามีการใช้งานทางเส้นนี้บ่อยๆ ก็จะทำให้ซินแนปติกส์สเตรงท์(หรือเวกซ์)แข็งแกร่งขึ้น (เซลล์สมองที่ใช้งานมากบ่อยๆ ก็จะทำให้ ซินแนปส์ ใหญ่ขึ้น การส่งผ่านข้อมูลพัลส์ไฟฟ้าทำได้ดีขึ้น ทำให้มีประสิทธิภาพดีขึ้น เช่นสามารถคิดหรือจดจำได้เร็วและดีขึ้น) โครงข่ายประสาทเทียมนี้ ใช้การเรียนรู้แบบเฮบบัน (Hebbian learning) จะเพิ่มค่าเวกซ์ของโครงข่ายอย่างสอดคล้องกับผลคูณของระดับความสนใจของแหล่งกำเนิดและจุดหมายของนิวรอล ตามสมการดังนี้

$$W_{ij}(n+1) = W_{ij}(n) + \alpha OUT_i OUT_j \dots\dots\dots(4.7)$$

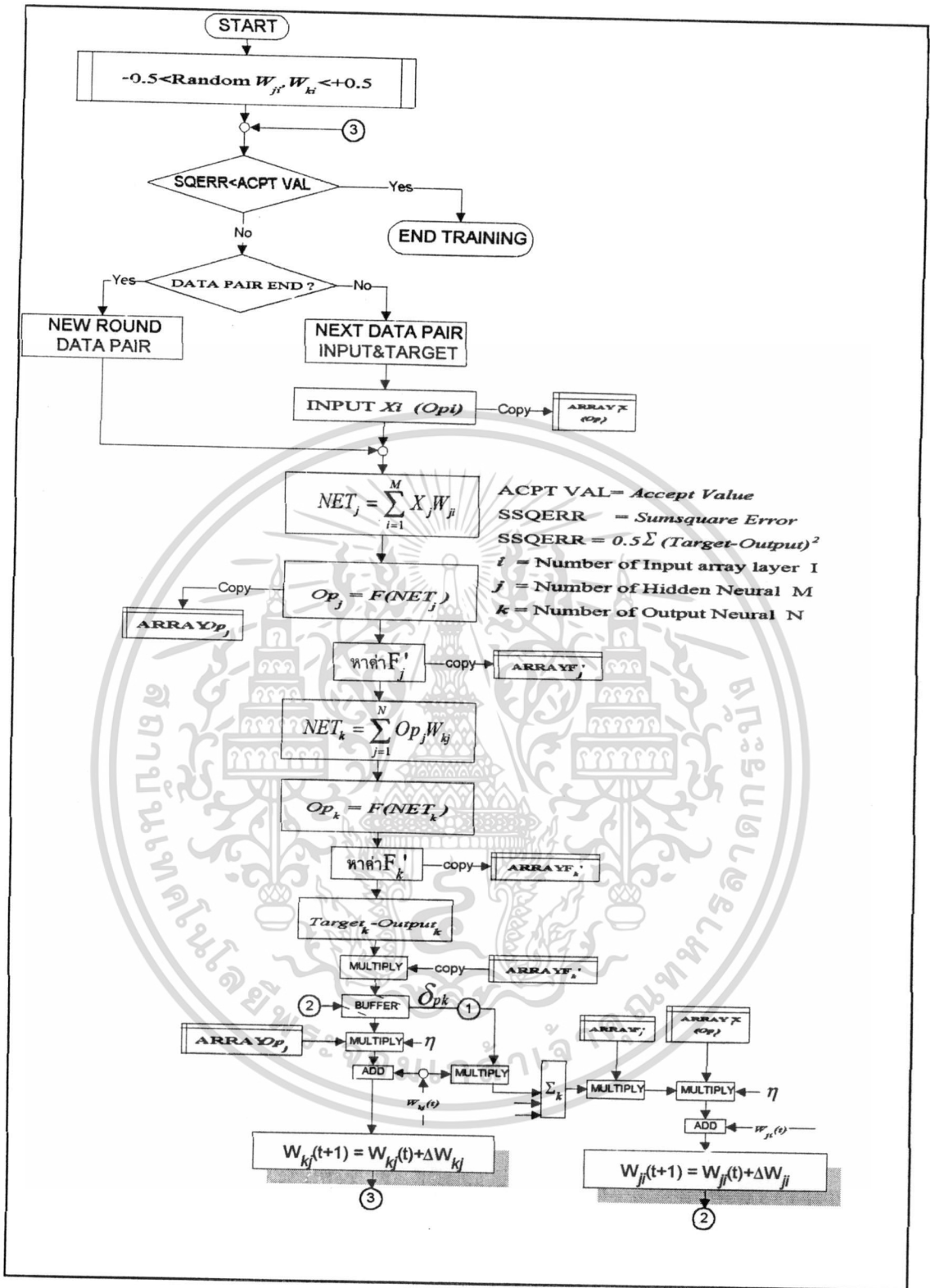
- โดย $W_{ij}(n)$ คือค่าเวกซ์ จากนิวรอล i ไปยังนิวรอล j ก่อนปรับปรุงค่า
- $W_{ij}(n+1)$ คือค่าเวกซ์ จากนิวรอล i ไปยังนิวรอล j หลังปรับปรุงค่า
- α คือค่าคงที่ของการเรียนรู้ (Learning rate coefficient)
- OUT_i คือเอาต์พุทของนิวรอล i และเป็นอินพุทของนิวรอล j
- OUT_j คือเอาต์พุทของนิวรอล j

โครงข่ายที่มีลักษณะการเทรนนิ่งแบบ เฮบเบิน นั้น เป็นผลมาจากการพัฒนามาแล้วกว่า 20-30 ปี โดยเฉพาะงานของโรเซนเบลท์ (Rosenblatt:1962), วิโดรว์(Widrow:1959), วิโดรว์ และฮอฟฟ์ (Widrow&Hoff:1960) และอีกหลายๆ คนที่พยายามพัฒนาระบบการเทรนนิ่งแบบควบคุม ที่สร้างโครงข่ายที่สามารถเรียนรู้แบบของอินพุตได้อย่างกว้างขวางและมีอัตราการเรียนรู้สูง ที่บรรลุผลได้จากหลักการพื้นฐานของการเรียนรู้หรือเทรนนิ่งให้กับโครงข่ายเช่น Perceptrons, Hopfieldnets, Backpropagation Networks และCounter propagation.



รูปที่ 5.8 แสดงไดอะแกรมแสดงวิธีการหาค่าเวกซ์ที่เหมาะสมของการเทรนนิ่งโดยใช้อัลกอริทึมแบบแบคพรอบพาเกชั่น

จากรูปที่ 5.8 และพอเขียนเป็น Flow Chart ได้ดังรูปที่ 5.9



รูปที่ 5.9 แสดง Flow chart แสดงกระบวนการเทรนนิ่ง เพื่อหาค่าเวกซ์ที่เหมาะสมในกระบวนการเทสดีง

5.7 เงื่อนไขการหยุดการฝึกสอน

การกำหนด เงื่อนไขการหยุดฝึกสอนทำได้สองกรณีคือ

1. เมื่อผลรวมของค่าผิดพลาดเฉลี่ยระหว่างเข้าที่พู่ทกับเป้าหมายทั้งหมด (SSQERR) ลดลง น้อยกว่าค่าที่กำหนด โดยกำหนดสมการหาค่าผิดพลาดเฉลี่ยของเข้าที่พู่ททั้งหมดดังนี้

$$SSQERR = \frac{1}{2} \sum (Target - Output)^2 \dots\dots\dots(4.20)$$

SSQERR คือ ค่า Sumsquare error ที่ได้จาก Output neurons เทียบกับ Target ใช้สำหรับบ่งชี้

ผลของการฝึกสอนว่าเพียงพอกับการใช้งาน หรือเงื่อนไขที่กำหนดหรือไม่

2. จำนวนรอบของการฝึกสอนดำเนินไปถึงค่าที่ตั้งไว้ ซึ่งจำนวนรอบที่เหมาะสมได้จากการทดลอง (ในการทดลองจำนวนรอบที่เหมาะสมอยู่ในช่วง 2,500-5,000 รอบ)



บทที่ 6

ขั้นตอนการทดลอง

งานวิจัยนี้ได้ทำการทดลองใน 2 ลักษณะ คือ

- 1.ลดขนาดข้อมูลเสียง ซึ่งกระทำกับข้อมูลเสียงทั้งในแกนเวลา และตามแกนความถี่
- 2.นำผลลัพธ์ที่ได้จากการลดขนาดข้อมูลเข้าสู่ขั้นตอนการรู้จำโดยใช้โครงข่ายประสาทเทียม

6.1 ประยุกต์การแปลงข้อมูลแบบคาร์ฮูเนนเลิฟ(Karhunen-Loeve Transform)ไปในการลดขนาดข้อมูลเสียง

การแปลงข้อมูลของคาร์ฮูเนนเลิฟ [4],[5] ลักษณะเด่นของวิธีนี้คือ สามารถลดขนาดของข้อมูลที่มีจำนวนมาก เช่น ข้อมูลภาพซึ่งจะมีข้อมูลที่ซ้ำซ้อนกันอยู่ วิธีนี้จะดึงเอาเฉพาะข้อมูลหลักที่เป็นตัวแทนของข้อมูลออกมาโดยตัดส่วนของข้อมูลที่ซ้ำซ้อนออกไป โดยที่ตัวแทนของข้อมูลเหล่านี้ เรียกว่าเวกเตอร์ สัมประสิทธิ์ และเวกเตอร์ไอเก้น จำนวนของข้อมูลที่จะเป็นตัวแทนนั้นขึ้นอยู่กับข้อกำหนดของงานประยุกต์แต่ละงาน โดยทั่วไปได้มาจากเวกเตอร์ไอเก้นที่มี ผลรวมของค่าไอเก้น ประมาณร้อยละ 95 ของ ผลรวมค่าไอเก้นทั้งหมด

การแปลงข้อมูลของคาร์ฮูเนนเลิฟ จะปรับข้อมูลเข้าให้อยู่ในรูปของเมตริกซ์ ที่มีจำนวนของแถวและคอลัมน์เท่ากัน โดยการหาโคแวนเรียนซ์เมตริกซ์ จาก สมการที่ 6.1

$$C_x = E\{(x - \bar{x})(x - \bar{x})^T\} \quad 6.1$$

เมื่อ $\bar{x} = E\{x\}$ คือเวกเตอร์เฉลี่ย ซึ่งคำนวณได้จากสมการ 6.2

$$\bar{x} \approx \frac{1}{M} \sum_{i=1}^M x_i \quad 6.2$$

ดังนั้น จาก 6.1 และ 6.2 จะได้

$$C_x \approx \frac{1}{M} \sum_{i=1}^M (x_i - \bar{x})(x_i - \bar{x})^T \quad 6.3$$

เมื่อ M คือขนาดของเวกเตอร์ข้อมูลเข้า กำหนดให้ Φ_i เมื่อ $i = 1, 2, \dots, n$ คือเวกเตอร์ไอเก้นของ C_x ที่คำนวณได้จากสมการ 6.4

$$C = \begin{bmatrix} c_{1,1} & c_{1,2} & \dots & c_{1,10} \\ c_{2,1} & c_{2,2} & & \\ \dots & & & \\ c_{10,1} & c_{10,2} & \dots & c_{10,10} \end{bmatrix}$$

จากสมการที่ 6.4 เราจะได้ค่าของไอเก้นจำนวน 10 ค่าตามแนวเส้นทะแยงมุมของเมทริกซ์นำค่าไอเก้นที่ได้มาจัดเรียงจากมากไปหาน้อยแล้วนำไปคำนวณหาเวกเตอร์ไอเก้น และเวกเตอร์สัมประสิทธิ์ ตามสมการที่ 6.5 ซึ่งต่อไปเราจะใช้ เป็นตัวแทนของข้อมูลเสียงโดยที่จำนวนของเวกเตอร์ไอเก้นมีขนาดเท่ากับจำนวนเฟรมของข้อมูล เราจะเลือกเวกเตอร์ไอเก้นสำหรับเป็นตัวแทนของข้อมูลโดย เลือกจากเวกเตอร์ไอเก้นที่มีค่าของผลรวมค่าไอเก้นเป็น ร้อยละ 95 ของผลรวมของค่าไอเก้นทั้งหมด กำหนดให้จำนวนที่เป็นไปตามเงื่อนไขดังกล่าวเท่ากับ k โดยที่ $k \leq N$ เมื่อแทนค่าใน สมการ 6.4 และ 6.5 จะได้เวกเตอร์ไอเก้น ขนาดเท่ากับ $N \times k$ และ เวกเตอร์สัมประสิทธิ์มีขนาดเท่ากับ $k \times P$ ขนาดของข้อมูลที่ลดลง (\bar{N}) คำนวณได้จากสมการข้างล่างนี้

$$\bar{N} = k(N+P) \quad 6.7$$

โดยที่ N คือ จำนวนเฟรมของข้อมูล

k คือ จำนวนของไอเก้นเวกเตอร์ที่เป็นไปตามเงื่อนไข

P คือ จำนวนของข้อมูลในแต่ละเฟรม

6.3 การประยุกต์ใช้วิธีการแปลงของคาร์สันเนลล์กับข้อมูลเสียงในแกนความถี่

การทดลองในส่วนนี้จะคล้ายกับการทดลองในข้อ 6.3 ต่างกันที่ข้อมูลเสียง จะต้องถูกแปลงจากแกนเวลาให้อยู่ในรูปของแกนความถี่เสียก่อนซึ่งในการวิจัยนี้ ได้เลือกวิธีการแปลงข้อมูลจากแกนเวลาให้อยู่ในแกนความถี่ ที่เรียกว่า การแปลงโคซายน์แบบไม่ต่อเนื่อง (Discrete Cosine Transform : DCT) ซึ่งวิธีนี้ มีลักษณะการทำงานคล้ายกับการแปลงฟูเรียร์อย่างรวดเร็ว (Fast Fourier Transform:FFT) สำหรับข้อดีของ DCT ที่มีต่อ FFT คือข้อมูลที่ได้จากการแปลงด้วยวิธี FFT จะให้สเปกโตรแกรมที่อยู่ในรูปของจำนวนเชิงซ้อน ซึ่งทำให้สเปกโตรแกรมที่ได้มีขนาดใหญ่ แต่วิธีการแปลงแบบ DCT จะให้สเปกตรัมที่อยู่ในรูปของจำนวนจริงเท่านั้น ด้วยเหตุนี้การแปลงข้อมูลแบบ DCT จะให้สเปกโตรแกรมที่มีขนาดเล็กกว่า และเมื่อนำไปแปลงกลับเพื่อให้ได้ข้อมูลเดิม ข้อมูลที่ได้จะไม่ผิดรูปไปจากเดิมมาก

การทดลองเริ่มจากนำข้อมูลเสียงที่อยู่ในแกนเวลาผ่านการแปลงโคซายน์ แบบไม่ต่อเนื่อง ดังสมการที่ 6.8

$$F(u,v) = \frac{4C(u)C(v)}{n^2} \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} f(j,k) \cos\left[\frac{(2j+1)u\pi}{2n}\right] \cos\left[\frac{(2k+1)v\pi}{2n}\right] \quad 6.8$$

การแปลงโคซายน์แบบไม่ต่อเนื่อง สามารถแปลงคืนกลับสู่ข้อมูลต้นฉบับได้ ดังสมการข้างล่างนี้

$$f(j,k) = \sum_{u=0}^{n-1} \sum_{v=0}^{n-1} C(u)C(v)F(u,v) \cos\left[\frac{(2j+1)u\pi}{2n}\right] \cos\left[\frac{(2k+1)v\pi}{2n}\right] \quad 6.9$$

ข้อมูลเมื่อนำมาผ่านการแปลงโคซายน์แบบไม่ต่อเนื่อง จะได้ข้อมูลในแกนระหว่าง ขนาดกับความถี่ จากนั้นเลือกข้อมูลร้อยละ 40 แรกของทั้งหมด(ปริมาณร้อยละ 40 ได้มาจากการทดลองแบบสุ่มหลายๆครั้ง แล้วพบว่า ค่าร้อยละ 40 ให้ผลการ ทดลองที่ดีที่สุด) มาเป็นข้อมูลเข้าสำหรับการแปลงแบบคาร์สุเนนเนลลิฟต่อไปดังสมการที่ 6.3-6.5 ซึ่งขนาดของข้อมูลที่ลดลงด้วยวิธีนี้ เป็นไปดังสมการ (10)

$$\bar{N} = k(0.4N+P) \quad 6.10$$

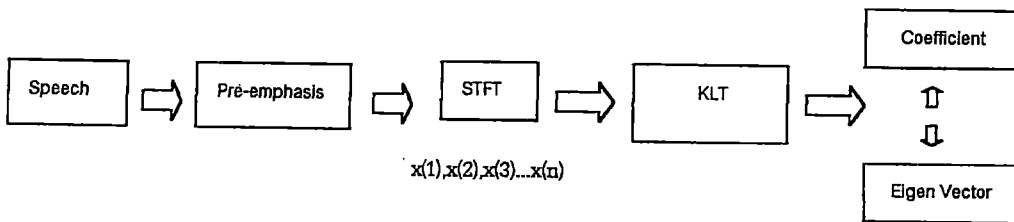
โดยที่ N คือ จำนวนเฟรมของข้อมูล

k คือ จำนวนของโอเก็นเวกเตอร์ที่เป็น ไปตามเงื่อนไข

P คือ จำนวนของข้อมูลในแต่ละเฟรม

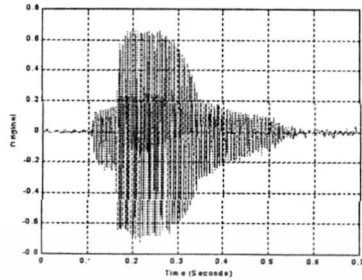
6.4 การนำผลลัพธ์ที่ได้จากการแปลงข้อมูลแบบคาร์สุเนนเนลลิฟ เข้าสู่ขั้นตอนการเรียนรู้โดยใช้โครงข่ายประสาทเทียม

ก่อนที่จะเข้าสู่ขั้นตอนการฝึกสอนและรู้จำ ข้อมูลเสียงจะต้องถูกเตรียมให้อยู่ในรูปแบบที่เหมาะสม โดยผ่านขั้นตอนดังแสดงในรูปที่ 6.1

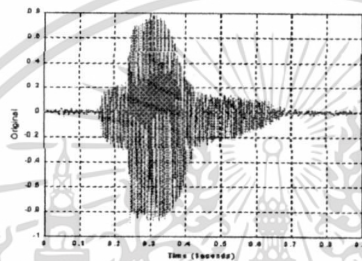


รูปที่ 6.1 แสดงโครงสร้างของระบบการเตรียมข้อมูล

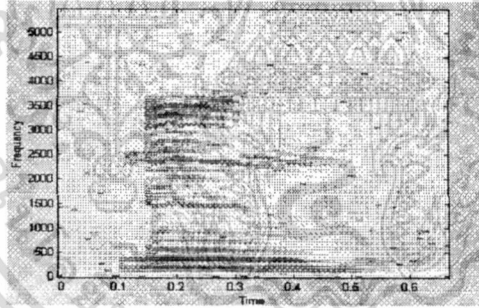
$$a(n) = \sqrt{\sum_{j=1}^J c^2(j,n)} \dots\dots 6.12$$



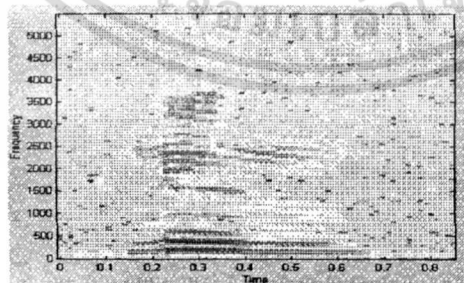
(a)



(b)



(c)



(d)

รูปที่ 6.2 (a),(b) เสียง ‘หนึ่ง’ ของผู้พูดคนที่ 1 และ 2 ตามแกนเวลา
 (c),(d) เสียง ‘หนึ่ง’ ของผู้พูดคนที่ 1 และ 2 ตามแกนความถี่

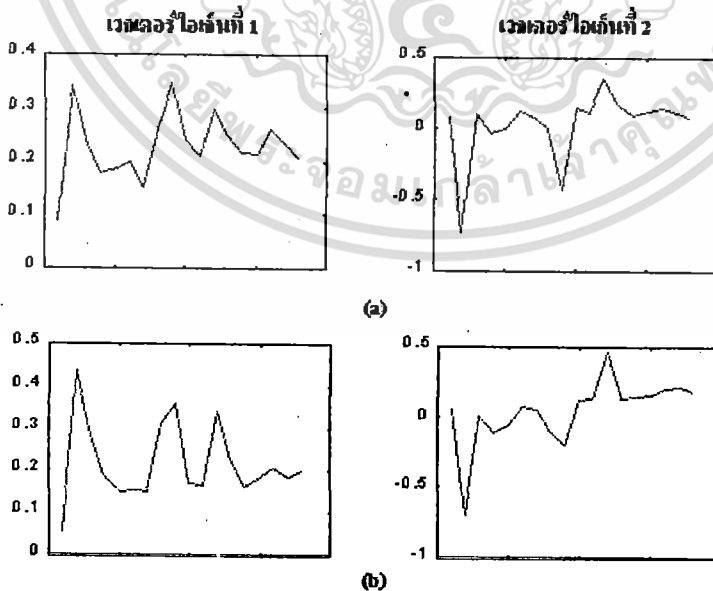
จากนั้นเข้าสู่ขั้นตอนการแปลงข้อมูล KLT โดยกำหนดให้ L คือ เวกเตอร์ซึ่งแต่ละเวกเตอร์มีจำนวนข้อมูลเท่ากับ J ($J=128$) การแปลงข้อมูล KLT นี้ ก็คือการคำนวณหาเวกเตอร์ไอเกน ซึ่งได้มาจาก โควาเรียนซ์เมตริก (Covariance Matix) แทนด้วย Σ มีขนาดเท่ากับ $J \times J$ โดยโควาเรียนซ์เมตริก ได้มาจาก 6.13

$$\Sigma = \frac{1}{L} \sum_{n=1}^L [p(n)p^T(n)] \dots\dots\dots 6.13$$

กำหนดให้ $\Phi = [\phi_1, \phi_2, \phi_3, \dots, \phi_J]$ คือ เวกเตอร์ไอเกนของ Σ แต่ละเวกเตอร์มีขนาดเท่ากับ $J \times 1$ คำนวณได้ตาม 6.14

$$\Sigma \Phi = \lambda \Phi \dots\dots\dots 6.14$$

λ คือ ค่าไอเกน (eigen value) = $\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_J$ โดยเรียงลำดับค่าจากมากไปหาน้อย สำหรับเวกเตอร์ไอเกน ที่จะเป็นตัวแทนของข้อมูลและเป็นข้อมูลที่แสดงลักษณะร่วมของข้อมูลที่ดีที่สุด ก็คือ ไอเกนเวกเตอร์ ที่ได้จากค่าไอเกนที่มีค่ามาก ซึ่งในงานวิจัยนี้จะเลือกไอเกนเวกเตอร์ที่ได้จากค่าไอเกนที่มากเป็นลำดับที่ 1 และ 2 ดังแสดงในรูปที่ 6.3 ซึ่งรูปที่ 6.3 แสดงเวกเตอร์ไอเกนที่ 1 และ 2 ของเสียง ‘หนึ่ง’ โดยที่ 3a แสดงเสียงของคนที่ 1 ครั้งที่ 1 และ 3b แสดงเสียงของคนที่ 1 ครั้งที่ 2



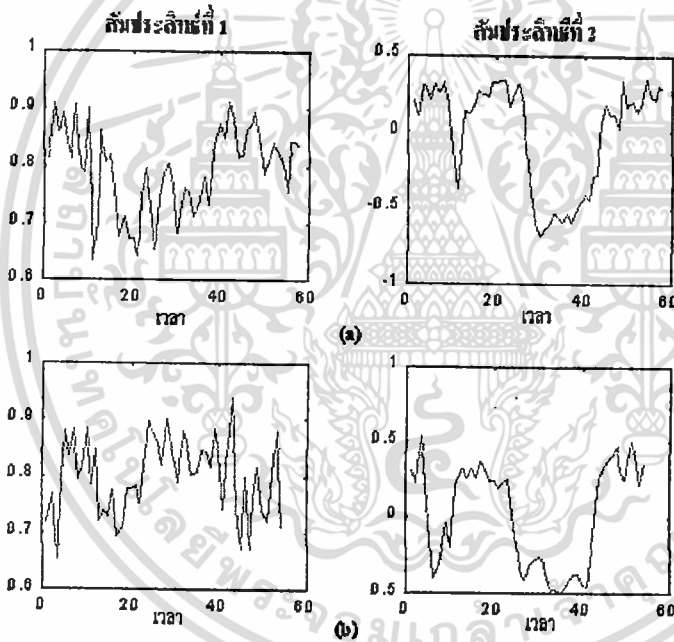
รูปที่ 6.3 แสดงเวกเตอร์ไอเกนที่ 1 และ 2 ของเสียง ‘หนึ่ง’

จากรูปที่ 6.3 จะเห็นได้ว่า เวกเตอร์ไอเกนที่ 1 และ 2 จะสามารถบอกลักษณะโครงสร้างสเปกตรัมของเสียงแต่ละเสียงได้ และยังสามารถวิเคราะห์ที่ ตำแหน่งของความถี่ฟอร์แมนต์ได้

ค่าสัมประสิทธิ์ของ KL (KL Coefficient) จะคำนวณได้โดยการถ่ายทอดข้อมูลจริงไปบนโดเมนใหม่ ซึ่งแทนได้โดยฟังก์ชันพื้นฐาน

$$y_j = \phi_j^T C \dots\dots 6.15$$

ซึ่งจะทำให้ได้ค่าเวกเตอร์สัมประสิทธิ์ (Coefficient Vector) y_j หนึ่งตัวมีความสัมพันธ์กับ เวกเตอร์ไอเกน (Eigen vector) หนึ่งชุด ซึ่งแสดงไว้ดังรูปที่ 4 แสดงสัมประสิทธิ์ คำว่า ‘หนึ่ง’ โดยที่ 6.4a แสดงเสียงของคนที่ 1 ครั้งที่ 1 และ 6.4b แสดงเสียงของคนที่ 1 ครั้งที่ 2 ค่าสัมประสิทธิ์ที่ได้ตามรูปที่ 4 บ่งบอกถึงการเปลี่ยนแปลงของสเปกตรัมที่แปรผันตามเวลา



รูปที่ 6.4 แสดงสัมประสิทธิ์ที่ 1 และ 2 ของเสียง ‘หนึ่ง’

บทที่ 7

ผลการทดลอง และสรุป

ในงานวิจัยนี้เราได้แบ่งการทดลองใน 2 งานวิจัย คือ ทดลองการลดขนาดข้อมูลโดยใช้การแปลงข้อมูลแบบคาร์สุ่มเน็นเลฟ และทดลองนำผลลัพธ์ของการลดขนาดข้อมูลทำการเรียนรู้โดยใช้โครงข่ายประสาทเทียม

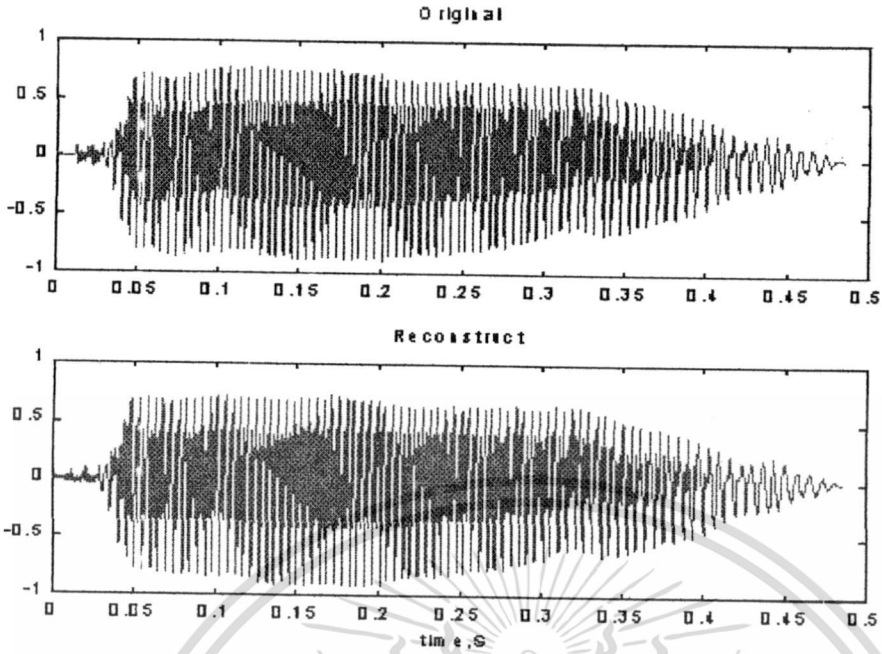
7.1 ทดลองการประยุกต์ใช้การแปลงข้อมูลแบบคาร์สุ่มเน็นเลฟลดขนาดข้อมูลเสียง ในงานวิจัยนี้ได้ทดลองกระทำใน 2 ลักษณะ ดังนี้

7.1.1 การแปลงข้อมูลของคาร์สุ่มเน็นเลฟในแกนเวลา

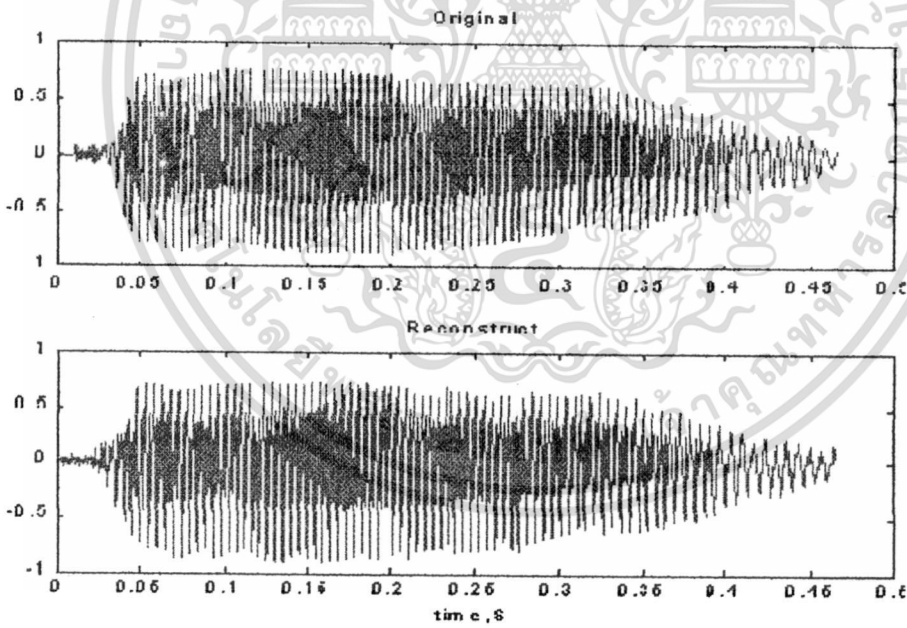
โดยจะใช้คำว่า ‘สวย’ เป็นตัวอย่าง แบ่งข้อมูลเสียงออกเป็นเฟรม ๆ ละ 256 ($P=256$) หรือ ประมาณ 0.025 วินาที จำนวนเฟรมที่ได้คือ N ($N=30$) จะได้ เมตริกซ์ D ขนาด $N \times P$ ทำเป็น โคเวเรียนซ์เมตริกซ์ (C มีขนาด $N \times N$) จากสมการ 6.3 จำนวนของค่าไอเก้น λ_i ทั้งหมด ที่ได้จากโคเวเรียนซ์เมตริกซ์ เท่ากับ N ($N=30$) นำค่าไอเก้นที่ได้จำนวน k ค่า โดยคิดเป็นร้อยละ 95 ของ ผลรวมค่าไอเก้นทั้งหมด $k = 16$ ค่า นำค่าไอเก้นที่ได้ ไปคำนวณหาเวกเตอร์ไอเก้นมีขนาดเท่ากับ $N \times k$ จากสมการ 6.5 หาเวกเตอร์สัมประสิทธิ์ y มีขนาดเท่ากับ $k \times P$ จากเวกเตอร์ไอเก้นและเวกเตอร์สัมประสิทธิ์ สามารถแปลงข้อมูลกลับได้ โดยสมการที่ 6.6 ในการหาคุณภาพของผลลัพธ์ที่ได้จากการแปลงข้อมูลกลับ เมื่อเทียบกับเสียงต้นฉบับ โดยการใช้การหาค่าผิดพลาดของกำลังสองเฉลี่ย (Mean Squared Error, MSE) และค่าของสัญญาณต่อสัญญาณรบกวน (Signal Noise Ratio, SNR) ผลการทดลองของคำว่า ‘สวย’ สามารถลดขนาด ข้อมูลลงไปได้ร้อยละ 40, $MSE=0.0003$, $SNR=16.36dB$ แสดงดังรูปที่ 7.1

7.1.2 การแปลงข้อมูลของคาร์สุ่มเน็นเลฟในแกนความถี่

ข้อมูลเสียงจะถูกแปลงให้อยู่ในรูปแกนความถี่ก่อนด้วยวิธีการแปลงโคซายน์ แบบไม่ต่อเนื่อง สมการที่ 6.8 จากนั้น นำข้อมูลที่ได้อ้อยละ 40 ค่าแรกของทั้งหมดมาผ่าน ขั้นตอนการแปลงข้อมูลของคาร์สุ่มเน็นเลฟ แปลงข้อมูลกลับด้วย สมการ 6.6 และ 6.9 แล้วเปรียบเทียบผลการทดลองที่ได้กับข้อมูลเสียงต้นฉบับโดยวิธีการหาค่าผิดพลาดของกำลังสองเฉลี่ยและอัตราส่วนระหว่างสัญญาณต่อสัญญาณรบกวน ผลการทดลองของคำว่า ‘สวย’ สามารถลดขนาดข้อมูลลงไปได้ร้อยละ 83, $MSE=0.0001$, $SNR=20.15dB$ แสดงดังรูปที่ 7.2



รูปที่ 7.1 ลักษณะของข้อมูลเสียงที่ผ่านการแปลงข้อมูลของคาร์ยูเน็นเลฟในแกนเวลา



รูปที่ 7.2 ลักษณะของข้อมูลเสียงที่ การแปลงข้อมูลของคาร์ยูเน็นเลฟในแกนความถี่

ผลการเปรียบเทียบการทดลองกับข้อมูลเสียงที่ใช้ในงานวิจัย ได้ผลแสดงตาม ตารางที่ 7.1

ข้อมูลเสียง	MSE	SNR (dB)	CR
แกนเวลา	0.0004	17.0232	ร้อยละ42
แกนความถี่	0.0003	19.1768	ร้อยละ79

ตารางที่ 7.1 แสดงผลการทดลองกับข้อมูลเสียงที่ใช้ในงานวิจัย

MSE = ค่าผิดพลาดของกำลังสองเฉลี่ย, SNR = ค่าของสัญญาณต่อสัญญาณรบกวน, CR = อัตราการลดขนาดของข้อมูล

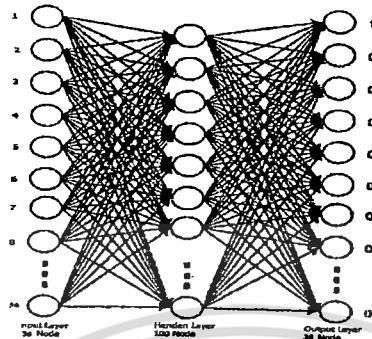
7.2 สรุปผลการทดลองการลดขนาดข้อมูลเสียง

งานวิจัยนี้ ได้นำเสนอวิธีการที่ใช้ในการลดขนาดข้อมูลเสียงที่ใช้ในการประมวลผลข้อมูล โดยใช้วิธีการแปลงข้อมูลของคาร์ยูเน็นเลิฟซึ่งสามารถประยุกต์ใช้ ในการลดขนาดข้อมูลเสียงทั้งที่อยู่ในแกนเวลาและแกนความถี่ได้ หลักการโดย รวมของวิธีของคาร์ยูเน็นเลิฟนี้ คือ การขจัดลักษณะของข้อมูลที่ซ้ำซ้อนกันให้เหลือ ข้อมูลที่เป็นลักษณะเด่นของเสียงเท่านั้นซึ่งลักษณะเด่นที่เป็นอิสระต่อกันนี้ เรียกว่า เวกเตอร์ไอเกนซึ่งจะถูกคัดเลือกมาจำนวนหนึ่งเป็นตัวแทนของข้อมูลเสียง การทดลองในงานวิจัยแบ่งออกเป็น 2 ส่วนคือการทดลองกับข้อมูลเสียงในแกนเวลาซึ่งตามผลการทดลองสามารถลดขนาดข้อมูลลงได้ร้อยละ 42 โดยมีค่าผิดพลาดกำลังสองเฉลี่ยเท่ากับ 0.0004 และอัตราส่วนของสัญญาณต่อสัญญาณรบกวนเท่ากับ 17.02dB และผลการทดลองเมื่อทดสอบกับข้อมูลเสียงในแกนความถี่ ซึ่งผ่านขั้นตอนการแปลงโคซายน์แบบไม่ต่อเนื่องมาแล้วได้ผลการทดลองเป็นร้อยละ 79 โดยมีค่าผิดพลาดกำลังสองเฉลี่ยเท่ากับ 0.0003 และอัตราส่วนของสัญญาณต่อสัญญาณรบกวนเท่ากับ 19.18dB

7.3 การทดลองนำผลลัพธ์ของการลดขนาดข้อมูลทำการเรียนรู้ และรู้จำโดยใช้โครงข่ายประสาทเทียม

ในงานวิจัยนี้ได้ทำการทดลองการเรียนรู้เสียงพูดภาษาไทยขึ้นอยู่กับผู้พูด กับเสียงพูดภาษาไทยจำนวน 20 คำ ดังนี้ ศูนย์-สิบ,เปิด,ปิด,หมุน,ยก,วาง,ช่อง,ซ้าย,ขวา และไฟ โดยเสียงจะมีความถี่อัตราการสุ่มเท่ากับ 11 kHz, 8 bit แบบ Mono ใช้ผู้พูดเพศชายอายุ 24 ปี 5 คน พูดคำละ 10 ครั้ง โดย 3 ครั้งแรกของแต่ละเสียงแต่ละคนใช้สำหรับการฝึกสอน โดยจะใช้เวกเตอร์ไอเกนทั้ง 2 เวกเตอร์ มาเรียงข้อมูลต่อกัน ดังนั้นข้อมูลอินพุตสำหรับขั้นตอนการฝึกสอนและทดสอบมีทั้งหมด 256

ข้อมูล (แต่ละเวกเตอร์ไอเก็นมี 128 ข้อมูล) ในงานวิจัยนี้ได้ใช้โครงข่ายประสาทเทียมแบบแพร่กระจายกลับ ซึ่งมีโครงสร้างและรายละเอียดของนิเวศน์คเวิร์คดังนี้



INPUT NEURONS = 256 MOMENTUM = 0.95
 OUTPUT NEURONS = 20 LEARNING RATE = 0.01
 HIDDEN NEURONS = 100 SSQERR = 0.01

โดยใช้เวลาในการฝึกสอนทั้งหมด 22 ชั่วโมง (ฝึกสอนบนเครื่องไมโครคอมพิวเตอร์ PENTIUM II ความเร็ว CPU = 233 MHz และขนาดของหน่วยความจำ = 64 MB) ในการทดสอบการรู้จำได้แบ่งผู้พูดออกเป็น 5 คน แต่ละคน พูด 20 คำ คำละ 10 ครั้ง รวมแล้วแต่ละคนจะพูดทั้งหมด 200 คำ ผลการทดลองอัตราการรู้จำของผู้พูดทั้ง 5 คน แสดงได้ตามตารางที่ 1

คนที่ทดสอบ	จำนวนข้อมูล	จำนวนที่จำถูก	%ความถูกต้อง
คนที่ 1	200	196	98
คนที่ 2	200	192	96
คนที่ 3	200	194	97
คนที่ 4	200	198	99
คนที่ 5	200	196	98
สรุป	1000	976	97.6

ตารางที่ 1 แสดงผลการทดลอง

7.4 สรุปผลการทดลองผลลัพธ์ของการบีบข้อมูลทำการเรียนรู้ และรู้จำโดยใช้โครงข่ายประสาทเทียม

งานวิจัยนี้ได้เสนอการประยุกต์ใช้วิธีการแปลงข้อมูลของคาร์สุเนนเลฟ (KLT) กระทำกับข้อมูลที่อยู่ในรูปของ Spectrum ผลที่ได้จากการกระทำนี้ เราสามารถวิเคราะห์โครงสร้างของสเปกตรัม ของแต่ละเสียงได้ และยังสามารถวิเคราะห์ลักษณะการเปลี่ยนแปลงของสเปกตรัมที่แปรผันตามเวลาของแต่ละเสียงด้วย จากการทดลองโดยนำเวกเตอร์ไอเกนที่มีค่าไอเกนมากเรียงตามลำดับ 2 ค่า ซึ่งก็เป็นการลดขนาดข้อมูลของข้อมูลเสียง โดยจะตัดข้อมูลที่มีความซับซ้อน เหลือแต่เฉพาะข้อมูล que แสดงลักษณะร่วมของ Spectrum ของเสียงไว้ จากนั้นนำข้อมูลนี้เป็นอินพุตให้กับโครงข่ายประสาทเทียมแบบแพร่กระจายกลับ (BPNN)

ในการรู้จำเสียงคำไทยจำนวน 20 คำ ระบบสามารถรู้จำได้ถูกต้องเฉลี่ยร้อยละ 97.6 ระยะเวลาในการฝึกสอนข้อมูลทั้งหมดเท่ากับ 22 ชั่วโมง ประมวลผลบนเครื่องไมโครคอมพิวเตอร์ PENTIUM II ความเร็ว CPU = 233 MHz และขนาดของหน่วยความจำ = 64 MB และ เวลาที่ใช้ในการรู้จำแต่ละคำเท่ากับ ~0.5 วินาที

อย่างไรก็ตามเสียงที่ทำให้เกิดการรู้จำผิดพลาดนั้น จากผลการวิเคราะห์แสดงให้เห็น ข้อมูลเสียงที่อยู่ในรูป Spectrum มีลักษณะแตกต่างกันถึงแม้จะเป็นเสียงจากคำ และผู้พูดคนเดียวก็ตาม ปัจจัยที่ก่อให้เกิดปัญหา อาจมาจาก ข้อมูลเสียงของผู้พูดแต่ละครั้งต่างกันซึ่งจำเป็นต้องมีการศึกษาเพิ่มเติมเกี่ยวกับ การรู้จำแบบ robustness และการปรับแบบจำลองของเสียง (model adaptation)

บทที่ 8

โปรแกรมการลดขนาดข้อมูล

8.1 โปรแกรมลดขนาดข้อมูลที่กระบนแกนเวลา

```

%%%%%% data compression and reconstruct using
Karhunen-Loeve method %%%%%%

```

```

clc
clear;

```

```

%%% load file .WAV %%%

```

```

name=input('filename.wav is ','s');
[Y,fs]=wavread(name);
sound(Y,fs);

```

```

%%% decompost frame=256 %%%

```

```

inpt=Y;
frame=256;
n_data=length(inpt);
n_windows=n_data/frame;
for j=1:n_windows
    for i=1:frame;
        z(j,i)=inpt(i+frame*(j-1));
    end;
end
inp=z;
[N,P] = size(inp);
power=z';

```

```

%%% Make zero mean %%%

```

```

[N,P]=size(power);
A=power';
B=mean(A);
mean_row=B';
for j=1:P;
    zz(:,j)=power(:,j)-mean_row;
end;
inp_zero=zz;

```

```

%%% covariance matrix %%%

```

```

cov=zz*zz';

%%% eigenvector form eigenvalue that has the
biggest eigenvalue = 10 %%%

[ei_vec,ei_va] = eig(cov);
[lambda,k]=sort(diag(ei_va));
lambda=lambda';
lambda=fliplr(lambda);
lambda=lambda';

%%% 95% of ei_va %%%

[n_lam,n_c]=size(lambda);
per_lam=0.97*sum(lambda);
n_ei=1;
temp=0;
for i=1:n_lam;

    if temp<=per_lam
        temp=temp+lambda(i,1);
        n_ei=n_ei+1;
    end;

end;
k=k(n_lam-n_ei+1:n_lam);
ei_vec=ei_vec(:,k);
M=n_ei;

%%% coefficients matrix %%%

coff=ei_vec'*inp_zero;

%%% reconstruct %%%

rec=ei_vec*coff;
for j=1:P;
    zzz(:,j)=rec(:,j)+mean_row;
end;
out=zzz;
x=out;
output=reshape(x,N*P,1);%%% end %%%
sound(output,fs);

```

```

%%%%%%%%%% demonstrate %%%%%%%%%%%
%%%%%%%%%%

```

```

x=power;
input=reshape(x,N*P,1);
x=out;
output=reshape(x,N*P,1);
nt=length(input);dt=1/fs;T=(nt-1)*dt;
tn=0:dt:T;
tn=tn';
figure(1);subplot(2,1,1);plot(tn,input)
title('Original')
figure(1);subplot(2,1,2);plot(tn,output)
title('Reconstruct');
xlabel('time,S')

```

```

%% Find error of Original and Reconstruct use
Root-mean-squared error(RMSE) %%

```

```

dif=(input-output).^2;
sum_dif=sum(dif);
mean=sum_dif/(N*P);
MSE=mean;
MSE_of_Original_Reconstruct = MSE

```

```

%% Find signal to noise ratio(SNR) %%

```

```

s_inp=input.^2;
s_nois=(input-output).^2;
sum_i=sum(s_inp);
sum_n=sum(s_nois);
SNR=10*log10(sum_i/sum_n);
i=reshape(input,N,P);
o=reshape(output,N,P);
i=i';
o=o';
s_inp=i.^2;
s_nois=(i-o).^2;
s_inp=sum(s_inp);
s_nois=sum(s_nois);
sub=s_inp./s_nois;
SSNR=10.*log10(sub);
SNR_of_Original_Reconstruct = SNR

```

```
%%% compression ratio %%%
```

```
rate=(n_data-M*(N+P))/(n_data)*100
```

8.2 โปรแกรมลดขนาดข้อมูลที่กระทำบนแกนความถี่

```
%%%%%%%% data compression and reconstruct using  
Karhunen-Loeve method %%%%
```

```
clc
```

```
clear
```

```
%%% load file .WAV %%%
```

```
name=input('filename.wav is ','s');
```

```
[x,fs]=wavread(name);
```

```
sound(x,fs);
```

```
%%% time to frequency domain using DCT %%%
```

```
[M1,N] = size(x);%z = zeros(M,N);
```

```
b = zeros(M1,N);
```

```
n_truncate=0.40*M1;
```

```
n_truncate=round(n_truncate);
```

```
y = dct(x);
```

```
figure(2); plot(y);
```

```
z = y(1:n_truncate);
```

```
%%% decompost frame=256 %%%
```

```
inpt=z;
```

```
frame=256;
```

```
n_data=length(inpt);
```

```
n_windows=n_data/frame;
```

```
for j=1:n_windows
```

```
    for i=1:frame;
```

```
        a(j,i)=inpt(i+frame*(j-1));
```

```
    end;
```

```
end
```

```
inp=a;
```

```
[N,P] = size(inp);
```

```
%%% covariance matrix %%%
```

```

cov=inp*inp';

%%% eigenvector form eigenvalue that has the
biggest eigenvalue = 10 %%%

[ei_vec,ei_va] = eig(cov);
[lambda,k]=sort(diag(ei_va));
lambda=lambda';
lambda=fliplr(lambda);
lambda=lambda';

%%% 95% of ei_va %%%

[n_lam,n_c]=size(lambda);
per_lam=0.97*sum(lambda);
n_ei=1;
temp=0;
for i=1:n_lam;
    if temp<=per_lam
        temp=temp+lambda(i,1);
        n_ei=n_ei+1;
    end;
end;
k=k(n_lam-n_ei+1:n_lam);
ei_vec=ei_vec(:,k);
M=n_ei;

%%% coefficients matrix %%%

coff=ei_vec'*inp;

%%% reconstruct %%%

rec=ei_vec*coff;
rec=rec';
in=reshape(rec,N*P,1);
b(1:N*P) = in;
xx = idct(b); %%% end %%%
sound(xx,fs);

%%%%%%%%% demonstrate %%%%%%%%%%
%%%%%%%%%

```

```

nt=length(x);dt=1/fs;T=(nt-1)*dt;
tn=0:dt:T;
tn=tn';
figure(1);subplot(2,1,1);plot(tn,x)
title('Original')
figure(1);subplot(2,1,2);plot(tn,xx)
title('Reconstruct');
xlabel('time,S')

%%% Find error of Original and Reconstrunc use
Root-mean-squared error(RMSE)  %%%

dif=(x-xx).^2;
sum_dif=sum(dif);
mean=sum_dif/(M1);
MSE=mean;
MSE_of_Original_Reconstrunc = MSE

%%% Find sinal to noise ratio(SNR) %%%

s_inp=x.^2;
s_nois=(x-xx).^2;
sum_i=sum(s_inp);
sum_n=sum(s_nois);
SNR=10*log10(sum_i/sum_n);
SNR_of_Original_Reconstrunc = SNR

%%% compression ratio %%%

rate=(M1-M*(N+P))/(M1)*100

```

เอกสารอ้างอิง

- [1] C.S. Chen and K.S. Huo : 'Karhunen-Loeve method for data compression and speech synthesis', IEE Proceeding-I, Vol 138, No. 5, October 1991, pp 377-380
- [2] Pree, William H., "Numerical Recipes in C", Cambridge University Press., 1988, pp 103-446
- [3] Deller, Jr., John, Froakis, John G., Hansen, John H.L., "Discrete-Time Processing of Speech Signals", Macmillan Publishing Company, 1993
- [4] K.I. Diamantaras, S.Y. Kung., "Principal Component Neural Networks : Theory and Applications" A wiley-Interscience Publication JOHN WILEY & SONS, INC.
- [5] JAIN, A.K. : 'A fast Karhunen-Loeve transform for a class of random processes', IEEE Trans., 1976, COM-24, pp. 1023-1029
- [6] V.Ralph Algazi, Kathy L.Brown, Michael J. Ready, David H.Irvine, Christie L. Cadwell and Sans Chung "Transform Representation of the Spectra of Acoustic Speech Segments with Application-I: General Approach and Application to Speech Recognition" IEE Transactions on speech and audio processing, VOL. 1, NO.2, April 1993