# การรู้จำแบบทนทานต่อเสียงรบกวนสำหรับเสียงพูดภาษาไทย โดยใช้อัลกอริทึม N-best LIMABEAM


โดย

นายรุ่งโรจน์ กอปัญญาพิพัฒน์


วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสตร์
ปีการศึกษา 2558

# การรู้จำแบบทนทานต่อเสียงรบกวนสำหรับเสียงพูดภาษาไทย โดยใช้อัลกอริทึม N-best LIMABEAM


โดย

นายรุ่งโรจน์ กอปัญญาพิพัฒน์




วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยธรรมศาสต
ปีการศึกษา 2558


# NOISE ROBUST SPEECH RECOGNITION FOR THAI LANGUAGE USING N-BEST LIMABEAM ALGORITHM

BY

MR. RUNGROJ KOPANYAPIPAT

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE
(COMPUTER SCIENCE)
DEPARTMENT OF COMPUTER SCIENCE
FACULTY OF SCIENCE AND TECHNOLOGY
THAMMASAT UNIVERSITY
ACADEMIC YEAR 2015

มหาวิทยาลัยธรรมศาสตร์

คณะวิทยาศาสตร์และเทคโนโลยี

วิทยานิพนธ์

ของ

นายรุ่งโรจน์ กอปัญญาพิพัฒน์

เรื่อง

การรู้จำแบบทนทานต่อเสียงรบกวนสำหรับเสียงพูดภาษาไทย
โดยใช้อัลกอริทึม N-best LIMABEAM

ได้รับการตรวจสอบและอนุมัติ ให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
วิทยาศาสตรมหาบัณฑิต (วิทยาการคอมพิวเตอร์)

เมื่อ วันที่ 14 กรกฎาคม พ.ศ. 2559

ประธานกรรมการสอบวิทยานิพนธ์ _____

(อาจารย์ ดร. เด่นดวง ประดับสุวรรณ)

กรรมการและอาจารย์ที่ปรึกษาวิทยานิพนธ์ _____

(ผู้ช่วยศาสตราจารย์ ดร. รัชฎา คงคะจันทร์)

กรรมการสอบวิทยานิพนธ์ _____

(อาจารย์ ดร. ปกรณ์ ลี้สุทธิพรชัย)

กรรมการสอบวิทยานิพนธ์ _____

(ผู้ช่วยศาสตราจารย์ ดร. เกรียงศักดิ์ เตมีย์)

คณบดี _____

(รองศาสตราจารย์ ปกรณ์ เสริมสุข)

| | |
|---|---|
| หัวข้อวิทยานิพนธ์ | การรู้จำแบบทนทานต่อเสียงรบกวนสำหรับเสียงพูดภาษาไทยโดยใช้อัลกอริทึม N-best LIMABEAM |
| ชื่อผู้เขียน | นายรุ่งโรจน์ กอปัญญาพิพัฒน์ |
| ชื่อปริญญา | วิทยาศาสตรมหาบัณฑิต |
| สาขาวิชา/คณะ/มหาวิทยาลัย | สาขาวิชาวิทยาการคอมพิวเตอร์<br>คณะวิทยาศาสตร์และเทคโนโลยี<br>มหาวิทยาลัยธรรมศาสตร์ |
| อาจารย์ที่ปรึกษาวิทยานิพนธ์ | ผู้ช่วยศาสตราจารย์ ดร. รัชฎา คงคะจันทร์ |
| อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม | - |
| ปีการศึกษา | 2558 |

## บทคัดย่อ

งานวิจัยนี้เสนอวิธีรู้จำเสียงพูดภาษาไทยแบบทนทานต่อเสียงรบกวนในสภาพแวดล้อมจริงที่ประกอบไปด้วยเสียงรบกวนจากเครื่องจักร และเสียงพูดที่ไม่สนใจ ซึ่งประสิทธิภาพในการรู้จำสามารถเพิ่มขึ้นได้ โดยการใช้หลักการของไมโครโฟนอาเรย์ (microphone array) ร่วมกับอัลกอริทึม N-best LIMABEAM ในการรู้จำเสียงพูด ซึ่งอัลกอริทึมนี้ใช้เพื่อคำนวณหาคำตอบจากการถอดรหัสเสียงหาค่าคุณลักษณะสำคัญของเสียงที่มีเสียงรบกวน ค่าสมมุติฐานจากการถอดรหัสเสียงที่ดีขึ้นในขั้นตอนการรู้จำในขั้นแรก แล้วหาค่าการถอดรหัสเสียงที่ดีที่สุดในแต่ละค่าสมมุติฐานจากการถอดรหัสเสียง ด้วยวิธีการจากอัลกอริทึม LIMABEAM แล้วสุดท้ายได้ผลลัพธ์ที่ดีที่สุดจากชุดค่าสมมุติฐานของ N-best จากผลการทดลองอัลกอริทึม N-best LIMABEAM ได้ค่าความถูกต้องอยู่ที่ 27.22% ซึ่งดีกว่าอัลกอริทึม LIMABEAM 20.12% และ เสียงที่มีเสียงรบกวน 9.47% ในสภาพแวดล้อมที่ถูกกำหนดเสียงรบกวน และ เสียงกังวาน

**คำสำคัญ**: การรู้จำเสียงพูดแบบทนทาน, เสียงพูดภาษาไทย, การประมวลผลเสียงพูด, การประมวลผลแบบหลายช่องรับสัญญาณ

| | |
|---|---|
| Thesis Title | Noise Robust Speech Recognition for Thai Language Using N-best LIMABEAM Algorithm |
| Author | Mr. Rungroj Kopanyapipat |
| Degree | Master of Science |
| Major Field/Faculty/University | Computer Science<br>Faculty of Science and Technology<br>Thammasat University |
| Thesis Advisor | Assistant Professor Rachada Kongkachandra |
| Thesis Co-Advisor (If any) | - |
| Academic Years | 2015 |


## ABSTRACT

In this thesis, I propose an approach for robust speech recognition in Thai language in everyday life environments. Performance of speech recognition can be increased by using a microphone array and N-best extension of the LIMABEAM algorithm is used for recognition. We show that this algorithm can be used to optimize the noisy acoustic features using the N-best hypothesized transcriptions generated at a first recognition step and then apply LIMABEAM algorithm in each N-best hypothesized transcriptions to get recognition result. The resulting N-best hypotheses list is automatically re-ranked. Results show improvements over LIMABEAM algorithm with considerable amount of noise and limited reverberation.



**Keywords:** Robust speech recognition, Thai speech, Speech processing, Microphone array processing.

# กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยดี เนื่องจากได้รับความกรุณาอย่างสูงจาก ผู้ช่วยศาสตราจารย์ ดร. รัชฎา คงคะจันทร์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่กรุณาสละเวลาอันมีค่าในการให้คำแนะนำปรึกษา ความรู้ ทักษะตลอดจนแนวทางในการปรับปรุงแก้ไขข้อบกพร่องต่าง ๆ ด้วยความเอาใจใส่อย่างดียิ่ง ผู้วิจัยตระหนักถึงความตั้งใจจริงและความทุ่มเทของอาจารย์ และขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ ที่นี้ และผู้วิจัยขอขอบพระคุณ ดร. เด่นดวง ประดับสุวรรณ ดร. ปกรณ์ ลี้สุทธิพรชัย และ ผู้ช่วยศาสตราจารย์ ดร. เกรียงศักดิ์ เตมีย์ กรรมการสอบวิทยานิพนธ์ที่ได้ให้คำปรึกษา ให้คำแนะนำ และเสียสละเวลาเพื่อตรวจสอบวิทยานิพนธ์ฉบับนี้

ผู้วิจัยขอขอบพระคุณคณาจารย์ภาควิชาวิทยาการคอมพิวเตอร์ทุกท่าน ที่ได้ประสิทธิ์ประสาทวิชาความรู้ให้แก่ผู้วิจัย เจ้าหน้าที่ภาควิชาวิทยาการคอมพิวเตอร์ทุกท่านที่อำนวยความสะดวกและให้คำแนะนำในการศึกษา รวมทั้งเพื่อน ๆ นักศึกษาปริญญาโททุกท่านที่ให้กำลังใจที่ดีตลอดมา

นอกจากนี้ผู้วิจัยขอขอบคุณหน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (http://tvis.nectec.or.th/speech) ที่เป็นแหล่งให้ความรู้เกี่ยวกับวิธีการรู้จำเสียงพูดภาษาไทย และข้อมูลเสียงพูด Corpus สำหรับใช้ในการศึกษาค้นคว้างานวิจัยฉบับนี้

สุดท้ายข้าพเจ้าขอขอบพระคุณบิดา มารดา ภรรยา บุตร และครอบครัวที่ให้กำลังใจอย่างดียิ่งจนทำให้วิทยานิพนธ์ฉบับนี้สำเร็จสมบูรณ์ได้ ขออำนาจคุณพระศรีรัตนตรัยและสิ่งศักดิ์สิทธิ์ทั้งหลาย โปรดดลบันดาลให้ท่าน จงประสบแต่ความสุข ความสำเร็จ สมบูรณ์ พูลลาภ ในสิ่งอันพึงปรารถนาทุกประการเทอญ

นายรุ่งโรจน์ กอปัญญาพิพัฒน์

มหาวิทยาลัยธรรมศาสตร์

พ.ศ. 2558

# สารบัญ

# สารบัญตาราง

ตารางที่ หน้า

# สารบัญภาพประกอบ

# บทที่ 1

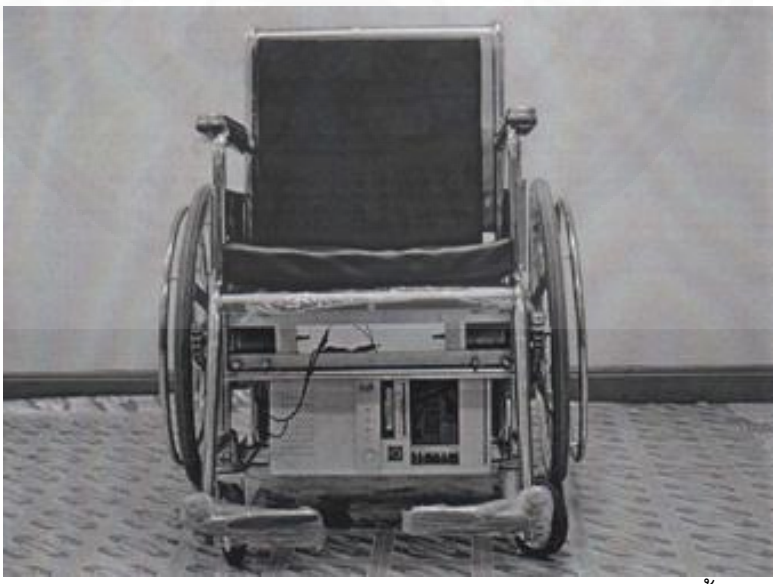
# บทนำ

## 1.1 ความเป็นมาและความสำคัญของปัญหา

ระบบรู้จำเสียงสำหรับภาษาไทย (Thai Speech Recognition) ได้มีการศึกษา และทำวิจัยมามากกว่า 40 ปีแล้ว ซึ่งปัจจุบันเทคโนโลยี Speech Recognition นี้มีประโยชน์อย่างมาก สามารถนำมาใช้ในชีวิตประจำวันได้หลากหลายอย่าง ดังตัวอย่างต่อไปนี้

1. ช่วยในการสั่งงาน รถเข็นคนพิการที่สามารถสั่งการได้ด้วยเสียงพูด (ณรงค์รัตน์ เลี้ยวรุ่งโรจน์, อนุพงษ์ ธรรมรักษาสิทธิ์, และ โกสินทร์ จำนงไทย, 2546) จะช่วยให้คนพิการเคลื่อนที่ไปในทิศทางต่าง ๆ ด้วยคำสั่งเสียงพูด ทำให้เกิดความสะดวกสบายในการเดินทางดังภาพที่ 1.1 คำสั่งเสียงพูดสำหรับควบคุมการใช้งานอุปกรณ์ไฟฟ้าภายในบ้าน ดังภาพที่ 1.2 เป็นต้น

ภาพที่ 1.1
รถเข็นคนพิการที่สามารถสั่งการได้ด้วยเสียง



ที่มา: “รถเข็นคนพิการควบคุมด้วยระบบรู้จำเสียงพูด, “ โดย ณรงค์รัตน์ เลี้ยวรุ่งโรจน์, อนุพงษ์ ธรรมรักษาสิทธิ์, และ โกสินทร์ จำนงไทย, จาก http://www.kmutt.ac.th/rippc/best47.htm

ภาพที่ 1.2

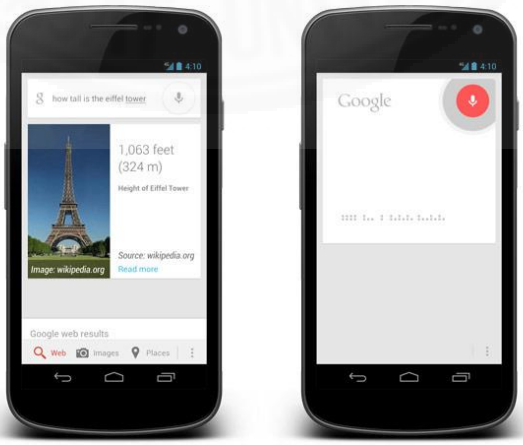
ควบคุมการใช้งานอุปกรณ์ไฟฟ้าด้วยเสียงพูด


ที่มา: “รายการสมรภูมิไอเดีย, “ จาก Internet

2. ช่วยในการค้นหาข้อมูล (Shan, Wu, Hu, Tang, Jansche, & Moreno, 2010, pp. 354-357) สามารถใช้เสียงพูดในการค้นหาข้อมูลที่ต้องการได้ ดังตัวอย่าง application ในโทรศัพท์ smartphone เช่น Google Now, S Voice (Samsung), Siri (Apple) เป็นต้น

ภาพที่ 1.3

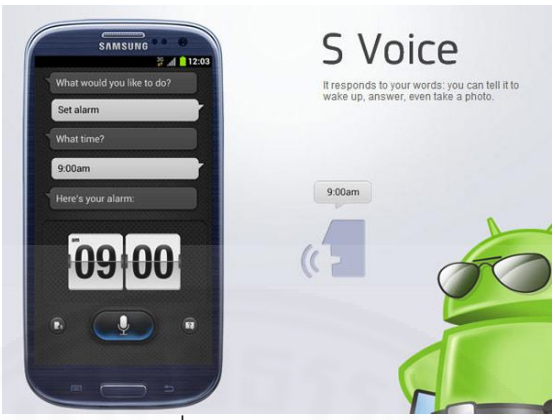
Google Now




ที่มา: จาก Internet

ภาพที่ 1.4

Samsung S Voice




ที่มา: จาก Internet

ภาพที่ 1.5

Apple Siri


ที่มา: จาก Internet

3. ช่วยให้ทำงานได้หลายอย่างพร้อมกัน เช่น ในขณะขับรถ สามารถสั่งให้โทรศัพท์โทรออกได้ด้วยเสียงพูดโดยที่ไม่ต้องใช้มือในการกดโทร ยังช่วยให้เกิดความปลอดภัยในการขับขี่รถด้วย ดังตัวอย่างโปรแกรม Dragon Drive ตามภาพที่ 1.6

ภาพที่ 1.6

โปรแกรม Dragon Drive


ที่มา: จาก http://www.nuance.com/

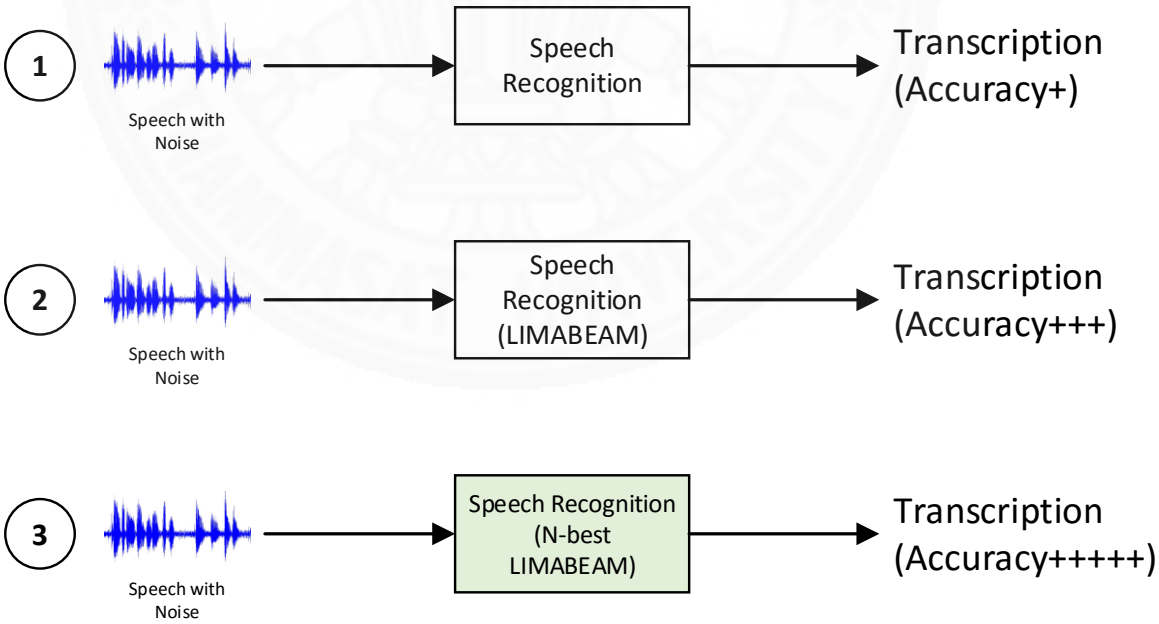
4. ลดอันตรายในการทำงาน เช่น ในโรงงานอุตสาหกรรม หรือ ในขณะขับรถ ช่วยให้ผู้ที่ปฏิบัติหน้าที่ในสถานการณ์ที่ไม่สามารถใช้วิธีการกด หรือพิมพ์เพื่อสั่งงานได้ หรือใช้ได้ไม่สะดวก โดยการใช้เสียงพูดในการสั่งการแทน

แต่อย่างไรก็ตามระบบรู้จำเสียงภาษาไทยที่มีอยู่ในปัจจุบัน ก็ยังทำงานได้ไม่ถูกต้อง 100% ซึ่งก็เกิดจากความผันแปรในด้านต่าง ๆ เช่น เสียงพูดที่ขึ้นอยู่กับผู้พูด ที่มีความแตกต่างในการพูดของแต่ละคนที่ไม่เหมือนกัน สำเนียงการพูดที่แตกต่างกันออกไปในแต่ละคน ผู้พูดที่เป็นชาย หรือ หญิง ก็จะให้เสียงที่แตกต่างกันออกไป ความผันแปรในด้านโดเมนของเรื่องที่จะพูด การพูดในกลุ่มของเรื่องที่ระบบเคยเรียนรู้ไปแล้วก็จะได้ผลที่แตกต่างกับการพูดในกลุ่มของเรื่องใหม่ที่ระบบยังไม่ได้เรียนรู้ ความผันแปรในด้านสภาพแวดล้อม ซึ่งในสภาพแวดล้อมที่แตกต่างกันก็จะให้ผลที่แตกต่างกันออกไป เช่น การใช้งานระบบรู้จำในสภาพแวดล้อมที่มีเสียงรบกวน อย่างเช่น เสียงแอร์ในห้อง เสียงเครื่องจักรทำงาน เสียงคนสนทนารอบข้าง เสียงรถยนต์ข้างถนน เสียงกังวานในห้อง ระยะห่างของผู้พูดกับเครื่องบันทึกเสียงที่อยู่ห่างเกินไป เป็นต้น ก็จะได้ความถูกต้องแตกต่างกันไป

เป็นที่รู้กันว่าระบบรู้จำนั้นยังทำงานได้ไม่ดีพอในสภาพแวดล้อมจริง ซึ่งเป็นสภาพแวดล้อมแบบเปิด ที่จะมีเสียงรบกวนรอบข้างตลอดเวลา ทั้งที่เป็นเสียงเครื่องจักร หรือ เสียงของผู้พูดที่เราไม่สนใจ ที่ผ่านมามีงานวิจัยทางด้านภาษาไทย เช่น (บุญเสริม กิจศิริกุล และ ณัฐกร ทับทอง, 2548, มนตรี โพธิโสโนทัย และ เฉลิมภัณฑ์ ฟองสมุทร, 2554) แต่ส่วนใหญ่จะเป็นแบบ single microphone และสามารถจัดการได้เฉพาะเสียงรบกวนที่เป็นเครื่องจักร

ดังนั้นงานวิจัยนี้จึงนำวิธีการรู้จำเสียงที่ทนทานต่อเสียงรบกวนใด ๆ หรือ เสียงที่ไม่ได้อยู่ในความสนใจ โดยการใช้อัลกอริทึม N-best LIMABEAM (Likelihood Maximizing Beamforming) ที่มีช่องทางในการรับเสียงมากกว่า 1 ช่องทาง แทนที่จะเป็นช่องทางเดียว และทำการเลือกผลลัพธ์ที่ดีที่สุดจาก N-best hypotheses list แทนที่จะใช้ผลลัพธ์ที่ดีสุดเพียงอันเดียว ซึ่งจะทำให้ได้ผลลัพธ์ที่ดีขึ้น จากการทดลองแสดงให้เห็นว่าอัลกอริทึม LIMABEAM สามารถพัฒนาให้ดีขึ้นได้ โดยสามารถสรุปเปรียบเทียบแนวทางการทำวิจัยได้ดังภาพที่ 1.7

ภาพที่ 1.7
การเปรียบเทียบแนวทางการทำวิจัย




ซึ่งจากภาพที่ 1.7 วิธีที่ 1 การรู้จำเสียงพูดแบบวิธีปกติที่ไม่มีอัลกอริทึมที่ช่วยในการรู้จำที่ทนทานต่อเสียงรบกวน ก็จะได้ผลลัพธ์ Transcription ที่มีค่าความถูกต้องน้อยที่สุด ส่วนวิธีที่

2 นั้นก็จะมีอัลกอริทึมที่ช่วยในการรู้จำแบบทนทานต่อเสียงรบกวน โดยใช้อัลกอริทึม LIMABEAM ซึ่งก็จะช่วยให้ได้ผลลัพธ์ Transcription ที่มีค่าความถูกต้องมากกว่าวิธีแรก ส่วนวิธีที่ 3 นั้นก็จะใช้อัลกอริทึมที่ทนทานต่อเสียงรบกวนเช่นกัน โดยใช้อัลกอริทึม N-best LIMABEAM ซึ่งเป็นอัลกอริทึมที่พัฒนาต่อยอดจากอัลกอริทึม LIMABEAM ก็จะช่วยให้ได้ผลลัพธ์ Transcription ที่มีค่าความถูกต้องมากที่สุดใน 3 วิธี

## 1.2 วัตถุประสงค์ของการวิจัย

1. ศึกษาการรู้จำแบบทนทานต่อเสียงรบกวนสำหรับเสียงพูด และนำมาประยุกต์ใช้สำหรับภาษาไทย
2. ศึกษาและประยุกต์ใช้อัลกอริทึม N-best LIMABEAM สำหรับการรู้จำเสียงพูดภาษาไทย
3. เปรียบเทียบประสิทธิภาพการรู้จำระหว่างอัลกอริทึม LIMABEAM กับ N-best LIMABEAM

## 1.3 ขอบเขตของงานวิจัย

สำหรับวิทยานิพนธ์ฉบับนี้ มีขอบเขตของงานวิจัยดังนี้

1. สภาพแวดล้อมที่ใช้ในงานวิจัย เป็นห้องทดลองขนาด 5.5 ม. x 4 ม. ที่มีเสียงรบกวนเป็นเสียงเครื่องปรับอากาศ และ เสียงโทรทัศน์
2. เป็นการรู้จำเฉพาะเสียงภาษาไทยแบบต่อเนื่องที่อยู่ในโดเมนของเรื่องการจองห้องพักโรงแรม มีจำนวนคำศัพท์ที่ใช้ 4,069 คำ
3. อัลกอริทึมที่ใช้ในการเปรียบเทียบประสิทธิภาพคือ LIMABEAM

## 1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถประยุกต์ใช้การรู้จำแบบทนทานต่อเสียงรบกวนสำหรับเสียงพูดภาษาไทย

2. สามารถประยุกต์ใช้อัลกอริทึม N-best LIMABEAM สำหรับการรู้จำเสียงพูดภาษาไทย

### 1.5 รายละเอียดวิทยานิพนธ์

รายละเอียดของวิทยานิพนธ์ฉบับนี้ ประกอบด้วยบทที่1 บทนำ จะกล่าวถึง ความเป็นมาและความสาคัญของงานวิจัยนี้ วัตถุประสงค์ ขอบเขตของงานวิจัย ประโยชน์ที่คาดว่าจะได้รับ บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง ประกอบไปด้วย ทฤษฎีที่เกี่ยวข้องกับการรู้จำเสียงพูด การรู้จำแบบทนทาน อัลกอริทึม LIMABEAM อัลกอริทึม N-best LIMABEAM รวมถึงงานวิจัยที่เกี่ยวข้อง บทที่ 3 การดำเนินงานวิจัย ประกอบไปด้วย ภาพรวมของระบบ สภาพแวดล้อมที่ใช้ในการทดลอง วิธีการดำเนินงานวิจัย ขั้นตอนการดำเนินงาน วิธีการประเมินผล บทที่ 4 ผลการวิจัยและอภิปรายผล ประกอบไปด้วย การทดลองและผลการทดลอง บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะสำหรับงานวิจัยในอนาคต

# บทที่ 2

# ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

## 2.1 ทฤษฎีและความรู้พื้นฐานที่เกี่ยวข้อง

### 2.1.1 Automatic Speech Recognition

ในงานวิทยานิพนธ์ฉบับนี้เป็นการพัฒนาวิธีการรู้จำเสียงพูดภาษาไทยแบบทนทาน โดยการใช้วิธีการรับสัญญาณเสียงจากชุด microphone array แล้วนำสัญญาณที่ได้ไปเข้าสู่กระบวนการรู้จำต่อไป ซึ่งวิธีนี้ก็จะช่วยให้การรู้จำมีความแม่นยำมากขึ้น ก่อนอื่นควรที่จะทำความเข้าใจวิธีการรู้จำเสียงพูดก่อนว่ามีวิธีดำเนินการอย่างไร โดยจะอธิบายวิธีการแปลงสัญญาณเสียงให้เป็นค่าคุณลักษณะสำคัญ (speech vector) ทำได้อย่างไร และการนำค่าคุณลักษณะสำคัญไปเข้าสู่กระบวนการรู้จำเพื่อให้ได้เป็นคำพูดได้อย่างไร เริ่มจากการอธิบายการได้มาด้วยค่าคุณลักษณะสำคัญด้วยวิธี mel-frequency cepstral coefficients (MFCC) หลังจากนั้นก็อธิบายระบบรู้จำด้วยวิธีของ Hidden Markov Models (HMM) ในการรู้จำโมเดลเสียง และการใช้โมเดลภาษาในการจัดโครงสร้างของภาษาให้เป็นรูปประโยค รวมทั้งบอกถึงหน่วยเสียงภาษาไทยที่มีการใช้งานตามหลักภาษา

#### 2.1.1.1 HMM-based Automatic Speech Recognition

ระบบการรู้จำเสียงพูดนั้นเป็นระบบที่ใช้ในการจัดประเภทของรูปแบบ (Rabiner & Juang, 1993) โดยตัวระบบจะจัดประเภทของสัญญาณเสียงที่อยู่ในรูปแบบของพยางค์ (phoneme) หรือคำ (word) ให้เป็นโมเดลโดยการจัดกลุ่มของคลาส (class) เป้าหมายของระบบรู้จำเสียงพูดนั้น คือการประเมินหาลำดับของคลาสที่ถูกต้องที่สุดที่ได้จากเสียงพูดแตกเป็นพยางค์จนประกอบเป็นคำ ดังแสดงเป็นสูตรคณิตศาสตร์ได้ดังนี้

$$\hat{w} = \arg\max_{w \in \mathcal{W}} P(\mathcal{Z}|w)P(w) \tag{2.1}$$

โดยที่ $\mathcal{Z}$ เป็น Observation sequence หรือ sequence ของ feature vector ที่ได้จากสัญญาณเสียง

$w$ คือ word sequence ของสัญญาณเสียง

$\mathcal{W}$ คือ ชุดของ word sequence ทั้งหมดที่เป็นไปได้จากระบบรู้จำ

$\hat{w}$ คือ ชุดของ word sequence ที่ได้จากระบบรู้จำ

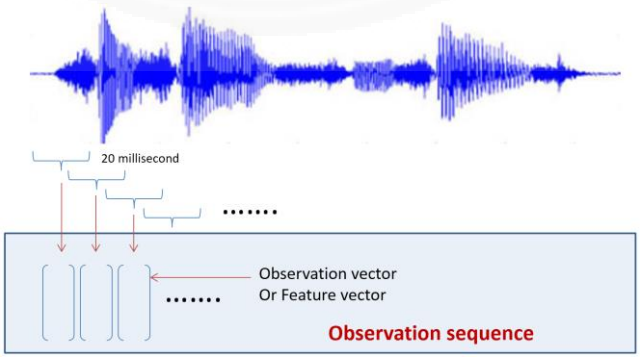
ดังนั้นกระบวนการในระบบรู้จำหลัก ๆ จะแบ่งเป็น 2 ส่วน คือ การสกัดหาค่าคุณลักษณะสำคัญ (Feature Extraction) ซึ่งก็เป็นการดึงค่าคุณลักษณะสำคัญของสัญญาณเสียงที่ได้ออกมาเป็น feature vector และอีกส่วนคือ การถอดรหัส (Decoding) ซึ่งเป็นการหา word sequence ที่เป็นไปได้มากที่สุดออกมา ซึ่งต่อไปก็จะอธิบายในรายละเอียดของการแปลงสัญญาณเสียงเป็นลำดับของ feature vector ด้วยวิธีแบบ MFCC จะทำอย่างไร และการนำเอา feature vector ไปทำการรู้จำโดยใช้ HMM-based จะมีวิธีการอย่างไร

2.1.1.2 **การหาค่าคุณลักษณะสำคัญ** (Feature Extraction)

ในระบบการรู้จำนั้นไม่ได้เป็นการนำเอาสัญญาณเสียงพูดมาเข้าสู่กระบวนการรู้จำโดยตรง แต่จะนำสัญญาณเสียงที่ได้มาทำการแบ่งสัญญาณออกเป็นส่วน ๆ ส่วนละสั้น ๆ ประมาณ 20 มิลลิวินาที เรียกว่า vector แล้วค่อย ๆ เลื่อน vector แบบ overlap กันไปทีละ 10 มิลลิวินาที เช่น หากมีเสียงยาว 1 วินาที ก็จะแทนด้วย vector ได้จำนวน 100 อัน Sequence ของ vector ที่แทนสัญญาณเสียงนี้จะเรียกว่า Observation sequence และจะเรียก vector ใด ๆ ใน Observation sequence ว่า Observation vector หรือ Feature vector นั่นเองดังภาพที่ 2.1

ภาพที่ 2.1

ภาพรวมการดึง Speech feature

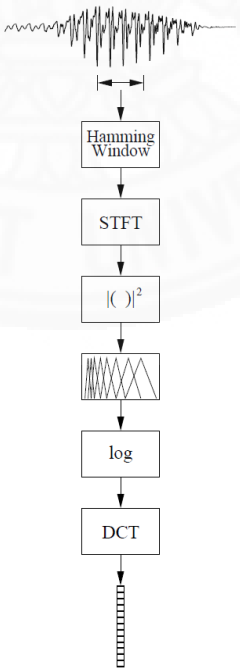



ที่มา: “Thai speech recognition tutorial [เว็บไซต์],” โดย หน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา, ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ

ในปัจจุบันระบบรู้จำส่วนใหญ่ จะใช้การหาค่าคุณลักษณะสำคัญด้วยวิธีของ mel-frequency cepstral coefficients (MFCC) หรือเรียกง่าย ๆ ว่า cepstral coefficients ซึ่งเป็นการพยายามที่จะประเมินค่าสัญญาณเสียงด้วยการใช้การคำณวณทางคณิตสาสตร์เป็นหลัก ซึ่งสัญญาณเสียงที่ได้มานั้นจะถูกแบ่งเป็นส่วน ๆ ส่วนละสั้น ๆ ทับซ้อนกันไปเรื่อย ๆ เรียกว่า frame หรือ vector ซึ่งในแต่ละ vector นั้นจะถูกดำเนินการในกระบวนการต่อไปนี้ แบ่งสัญญาณเสียงเป็น frame โดยใช้ Windows อัลกอริทึม แล้วแปลงเป็น frequency domain โดยใช้วิธี Short-Time Fourier Transform (STFT) เมื่อคำนวณค่า STFT แล้วจะนำไปคูณกับชุดของ triangular weighting ที่ทับซ้อนกัน เรียกว่า mel filters ค่า filter ที่ได้จะถูกคำนวณต่อด้วยการประเมินแบบ linearly ที่ช่วงความถี่ต่ำ และแบบ logarithmically ที่ช่วงความถี่สูง mel spectrum ของ frame ที่ถูกคำนวณได้ก็คือ vector ที่แสดงถึงค่าของแต่ละ mel filters สุดท้าย vector จะถูกแปลงไปเป็น mel-frequency cepstral โดยผ่าน Discrete –Cosine Transform (DCT) กระบวนการของการหาค่าคุณลักษณะสำคัญ (Feature Extraction) สามารถแสดงได้ดังภาพที่ 2.2

ภาพที่ 2.2

กระบวนการหาค่าคุณลักษณะสำคัญ (Feature Extraction)



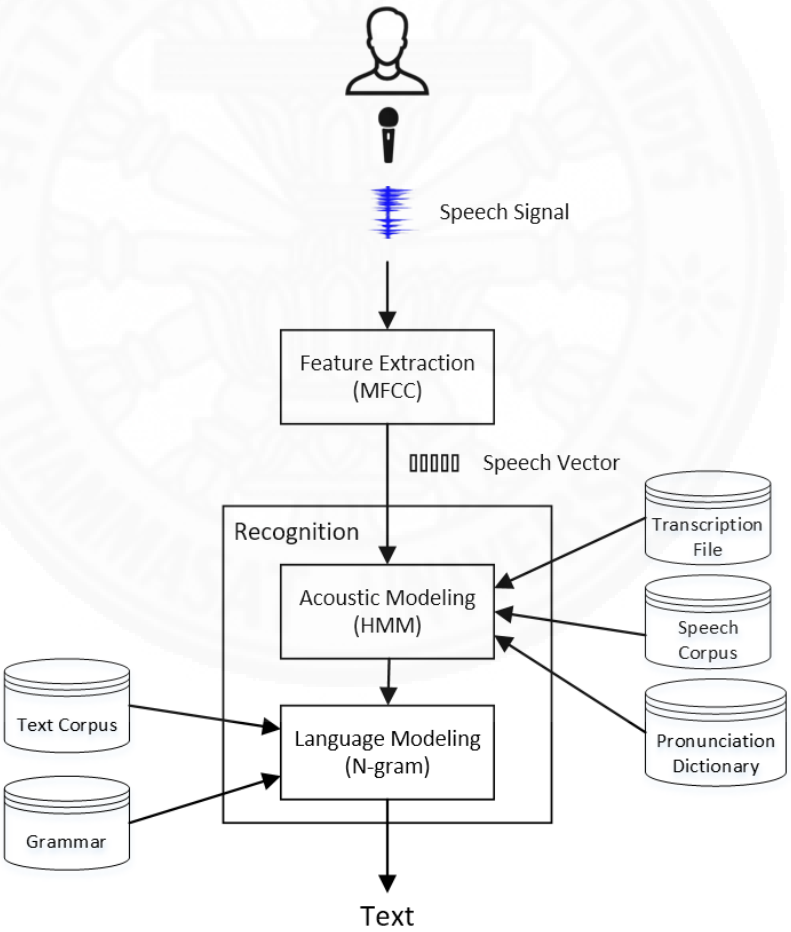

ที่มา: “Microphone Array Processing for Robust Speech Recognition,” by Seltzer, M. L., 2003, p. 10

2.1.1.3 **หลักการรู้จำ** (Speech Recognition)

ภาพรวมของกระบวนการรู้จำนั้นสามารถแสดงได้ดังภาพที่ 2.3 โดยกระบวนการหลัก ๆ เริ่มจากการนำสัญญาณเสียงที่ได้นำเข้าสู่กระบวนการหาค่าคุณลักษณะสำคัญ (Feature Extraction) แล้วจะได้เป็น Feature vector แล้วนำเข้าสู่กระบวนการรู้จำ ซึ่งในกระบวนรู้จำนี้ก็จะนำ Feature vector ไปผ่านโมเดลเสียง (Acoustic Model) เพื่อให้ได้ความน่าจะเป็นของ Phone เสียง แล้วเข้าสู่โมเดลภาษา (Language Model) เพื่อให้ได้เป็นโครงสร้างของคำ (Word)

ภาพที่ 2.3

ภาพรวมกระบวนการรู้จำ



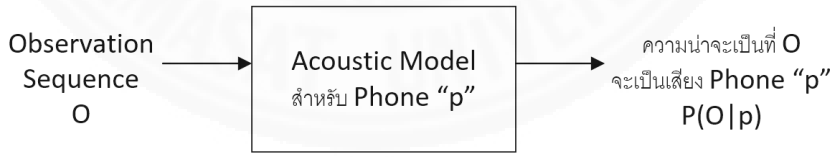
สิ่งที่ควรจะต้องรู้เพิ่มเติมเกี่ยวกับสิ่งที่ใช้ในกระบวนการรู้จำ ควรจะต้องรู้จักคำว่า Phone คำว่าโมเดลการออกเสียง (Pronunciation Model) คำว่าโมเดลเสียง (Acoustic Model) คำว่าโมเดลภาษา (Language Model)

Phone คือ หน่วยย่อยสุดทางเสียง ตัวอย่างเช่นคำว่า “การ” อ่านออกเสียงด้วยเสียง “ก” ตามด้วยสระ “า” และลงท้ายด้วยเสียงตัวสะกด “น” ในทางภาษาศาสตร์ จะมีสัญลักษณ์มาตรฐานแทนเสียง Phone แต่ละเสียง ตัวอย่างเช่น “k” แทนเสียง “ก” “aa” แทนสระ “า”

โมเดลการออกเสียง (Pronunciation Model) คือ มีหน้าที่ในการบอกว่าคำ (Word) ใด ๆ ออกเสียงอย่างไร โดยจะบอกเป็น sequence ของ Phone เช่น “การ” ออกเสียงว่า “k aa n” “ขนม” ออกเสียงว่า “kh a n o m^” วิธีการสร้างโมเดลการออกเสียงง่าย ๆ ก็คือการสร้าง Pronunciation Dictionary ซึ่งเป็นที่เก็บ list ของคำคู่กับเสียงอ่าน

โมเดลเสียง (Acoustic Model) คือ โมเดลที่ใช้ในการบอกว่าเมื่อมี Observation sequence เข้าไปยังโมเดลเสียงใด ๆ จะมีการคำนวณค่าความน่าจะเป็นที่ Observation sequence นั้นจะเป็นเสียงของ Phone นั้น ๆ ความน่าจะเป็นสรุปได้เป็น P(O|p) โดยที่ p คือโมเดลเสียงของ Phone ใด ๆ โดยปกติจะมีโมเดลเสียง 1 โมเดลต่อ 1 Phone สามารถสรุปได้ดังภาพที่ 2.4

ภาพที่ 2.4
หน้าที่ของ Acoustic Model



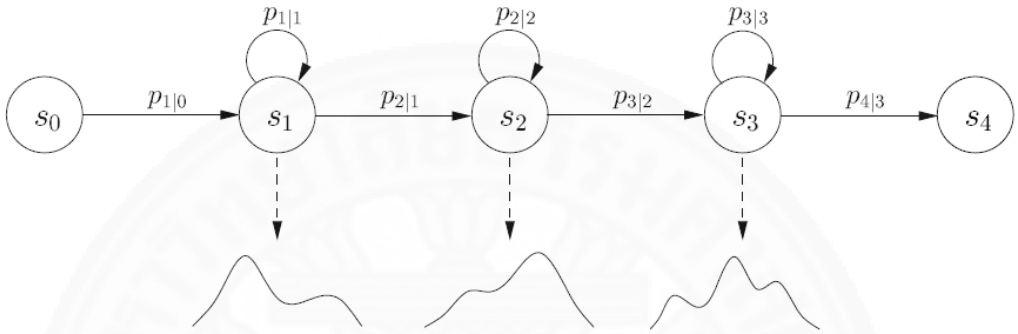

ที่มา: “Thai speech recognition tutorial [เว็บไซต์],” โดย หน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา, ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ

Acoustic Model ที่นิยมใช้ในปัจจุบัน คือ Hidden Markov Model (HMM) ใช้เป็นรูปแบบมาตรฐานสำหรับใช้แทนเสียง Phone 1 Phone ดังแสดงได้ดังภาพที่ 2.5 การทำงานนั้น ก็จะนำ Observation sequence เข้าทางซ้าย หรือเข้าทาง Node ที่ 1 และออกทาง Node สุดท้าย

ในระหว่างที่ Observation vector แต่ละตัววิ่งอยู่ใน HMM นี้ จะวิ่งไปข้างหน้า หรือย่ำอยู่ที่ Node เดิม (ตาม link ที่มีอยู่) เท่านั้น ไม่มีการย้อนกลับ จึงเรียก HMM แบบนี้ว่า Left-to-right Model

ภาพที่ 2.5

left-to-right HMM แบบ 5-state



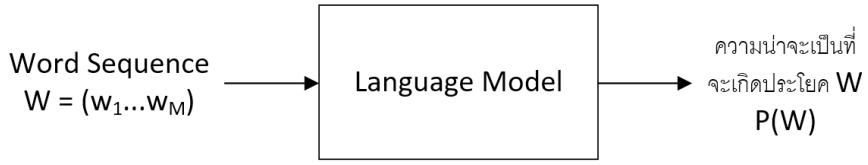

ที่มา: “Microphone Array Processing for Robust Speech Recognition,” by Seltzer, M. L., 2003, p. 11

Link ที่เชื่อมระหว่าง Node เรียกว่า Transition และ Node แต่ละอันเรียกว่า State ในขณะที่ Observation vector หนึ่ง ๆ กระโดดจาก State i ไป State j จะมีความน่าจะเป็นของการกระโดดคือ $p_{ij}$ เราเรียกความน่าจะเป็นของการกระโดดนี้ว่า Transition Probability และเมื่อ Observation vector ไปตกที่ State i จะมีความน่าจะเป็นของการตกคือ bi(O) ซึ่งเรียกว่า Emission Probability

โมเดลภาษา (Language Model) คือ จะเป็นตัวบอกว่าคำ (Word) นี้ ตามด้วยคำนี้ได้หรือไม่ หรือในบางโมเดลจะบอกค่าความน่าจะเป็นที่คำใด ๆ จะพูดต่อกันเช่น โมเดลภาษาอาจจะบอกว่า “จะ ไป” มีโอกาสเกิดได้แค่ 0.01 เป็นต้น โมเดลภาษาแท้จริงไม่เพียงบอกโอกาสที่คำสองคำจะเกิดคู่กันเท่านั้น ยังสามารถบอกด้วยว่า ทั้งประโยคมีโอกาสเกิดได้เท่าไหร่ สมมติว่าเรามีประโยคซึ่งประกอบด้วยคำต่อ ๆ กันหลาย ๆ คำ ขอแทนด้วย W คือ W = (w1…wM) โดยที่ w แทนคำแต่ละคำ โมเดลภาษาจะบอกว่า W สามารถเกิดได้หรือไม่ หรือบอกเป็นค่าความน่าจะเป็นว่ามีโอกาสเกิดมากน้อยแค่ไหน ค่าความน่าจะเป็นแทนด้วย P(W) สามารถสรุปได้ดังภาพที่ 2.6

ภาพที่ 2.6

หน้าที่ของ Language Model




ที่มา: “Thai speech recognition tutorial [เว็บไซต์],” โดย หน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา, ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ

การรู้จำจะทำตามขั้นตอนโดยละเอียดดังต่อไปนี้

1. ระบบรับ Observation Sequence ที่ต้องการรู้จำเข้ามา

2. ระบบจะเริ่มเดาว่าเป็นคำใดต่อ ๆ กัน เช่น จะเป็นประโยคว่า “ฉัน รัก เธอ” ประโยคว่า “ฉัน หิว ข้าว” หรือ “อาหาร อร่อย ดี” ฯลฯ

3. หลังจากเดาประโยคขึ้นมาแล้ว จะส่งประโยคนั้นเข้าไปยัง Language Model ก็จะได้ค่าความน่าจะเป็น P(W) ที่จะเกิดประโยคดังกล่าว

4. ทำการแปลงประโยคเป็นเสียงอ่านโดยอาศัย Pronunciation Model เช่น จาก “ฉัน รัก เธอ” เป็น “ch a n^ r a k^ th vv”

5. เมื่อได้ Sequence ของ Phone แล้ว ก็จะเอา Acoustic Model ของแต่ละ Phone มาต่อกัน แล้วทำการป้อน Observation Sequence เข้าไปยัง Acoustic Model ของทั้งประโยค (ดังนั้น Acoustic Model จะต้องสามารถต่อกันได้) ก็จะได้ค่าความน่าจะเป็น P(O|W) ซึ่งเกิดจาก P(O|p) ของแต่ละ Phone คูณกัน

6. สุดท้ายก็จะเอา P(W) ในข้อ 3 มาคูณกับ P(O|W) ในข้อ 5 ได้เป็น P(O, W) ซึ่งหมายถึง โอกาสที่สัญญาณเสียงดังกล่าวจะเป็นเสียงประโยค W

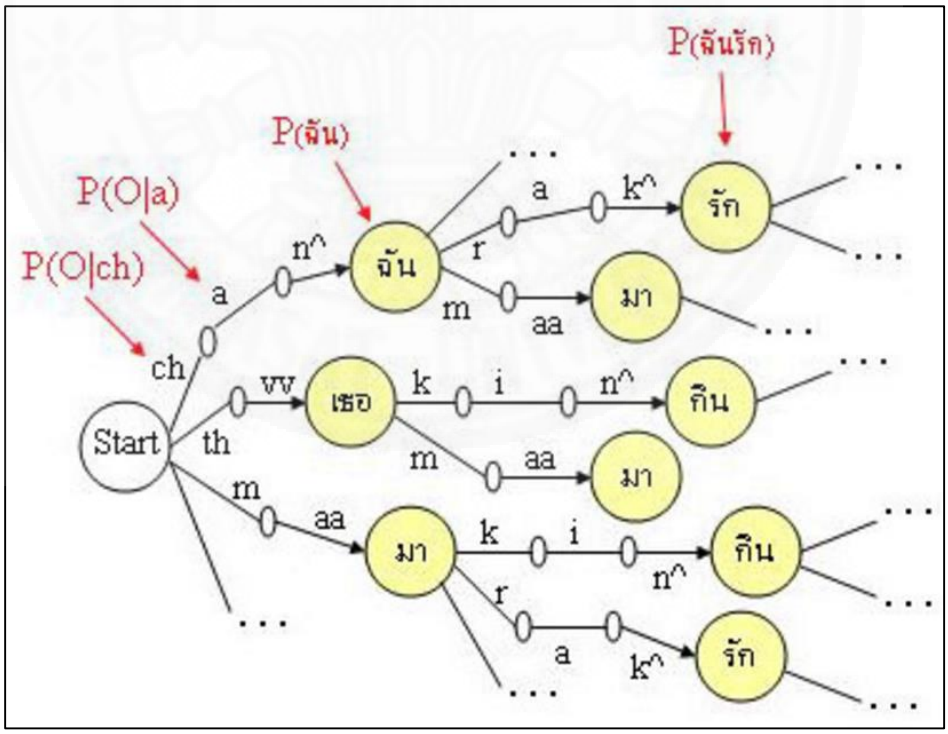
7. ทำอย่างนี้กับทุกประโยคที่เดาขึ้นมา และเทียบค่า P(O, W) ว่าประโยคไหนมีโอกาสสูงที่สุด ก็ตอบเป็นประโยคนั้น

การรู้จำจะมีปัญหาเกิดขึ้นเกี่ยวกับรูปแบบของประโยคที่ต้องเดาที่มีจำนวนมาก เนื่องจากประโยคที่เป็นไปได้มีหลากหลายแบบ ยิ่งไม่กำหนดว่าประโยคยาวเท่าไหร่ด้วยแล้ว จะมีประโยคที่เป็นไปได้จำนวนไม่สิ้นสุดเลย ซึ่งก็จะทำให้ระบบทำงานไม่ได้แน่นอน วิธีการแก้ไข คือ

การสร้าง Word Network โดยเอาคำมาต่อ ๆ กันในลักษณะของ Network ดังภาพที่ 2.7 ระหว่างคำก็กำกับด้วยโอกาสที่แต่ละคำจะต่อกัน หรือ P(wi|wi-1) และในแต่ละคำก็ประกอบด้วย Acoustic Model ของ Phone ที่ต่อกันเป็นเสียงอ่านของคำนั้น ๆ เวลาทำงานก็จะผ่านสัญญาณเสียงเข้าไป ในขณะที่ผ่าน Node ของ Network แต่ละ Node ก็จะมีการคูณค่าความน่าจะเป็น P(O, W) ต่อ ๆ ไปเรื่อย ๆ หากเส้นทางใดมีค่าความน่าจะเป็นรวมขณะนั้น ต่ำกว่าค่า Threshold ที่กำหนดไว้ ก็ให้เลิกวิ่งไปเส้นทางนั้น เท่านี้ก็จะช่วยลดจำนวนประโยคที่จะต้องคำนวณลงได้มากมาย วิธีนี้เรียกว่า Beam Search ก็คือ Search ภายใน Beam ที่กำหนดเท่านั้น ยังมีอีกวิธีในการกำหนด Beam ของการ Search โดยกำหนดให้ ณ ขณะใด ๆ จะมีเส้นทางที่วิ่งไปได้ไม่เกิน N เส้นทาง วิธีนี้ก็จะช่วยลดจำนวนประโยคที่ต้องคำนวณลงมากเช่นกัน เรียกวิธีนี้ว่า N-best Search

ภาพที่ 2.7

Word Network สำหรับการรู้จำ




ที่มา: “Thai speech recognition tutorial [เว็บไซต์],” โดย หน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา, ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ

ระบบรู้จำแท้ที่จริงแล้วก็คือการทำหน้าที่ Search ไปตาม Word Network หรือเรียกการทำหน้าที่ส่วนนี้ว่า Decoder ก็ได้ สรุปแล้วการรู้จำประกอบด้วย Decoder ที่ทำหน้าที่สร้าง Word Network และ Search ไปตามเส้นทางบน Network โดยอาศัย Acoustic Model และ Language Model เป็นตัวคำนวณความน่าจะเป็นของการเกิด Phone และ Word นั่นเอง

**2.1.1.4 หน่วยเสียงภาษาไทย**

นิยามหน่วยเสียงสำหรับกำหนดให้เสียงพยางค์ภาษาไทย มีจำนวนหน่วยเสียงเดี่ยว 68 หน่วยเสียง มีรายละเอียดดังนี้

ตารางที่ 2.1

หน่วยเสียงภาษาไทย

| พยัญชนะต้น (Ci) | | | |
|---|---|---|---|
| เดี่ยว | ตัวอย่าง | ผสม | ตัวอย่าง |
| p | ปาก | pr | ประสาน |
| t | เต้น, กฎิ | phr | พราน |
| c | จะ | tr | เตรียม |
| k | ก่อน | kr | กราบ |
| z | อาน | khr | คร่า |
| ph | พบ, ภัย, ผ่าน | pl | ปลา |
| th | ทิ้ง, ธง, เฒ่า, ฐาน, มณโฑ | phl | พลาด |
| ch | ชอบ, เฌอ | thr | จันทรา |
| kh | คน, เขิน, ฆ่า | kl | เกลอ |
| b | บอก | khl | เคลื่อน |
| d | ด้าน, ฎฎา | kw | กวาง |
| m | ไม่ | khw | ขวา |
| n | นาน, เณร | | |
| ng | เงิน | เสียงทับศัพท์ | |
| l | เล่น, กีฬา | br | เบรน |
| r | รอ, ฤทัย | bl | บลู |
| f | ฝน, ฟัน | fr | ฟราย |
| s | สาย, ศิลา, รักษา, ซ่อน | fl | เฟลม |
| h | โหน, เฮฮา | dr | ดราก้อน |
| w | ว่า | | |
| j | ย้อน, หญิง | 17 หน่วย | |
| 21 หน่วย | | | |

| สระ (V) | | | |
|---|---|---|---|
| เดี่ยว | ตัวอย่าง | ผสม | ตัวอย่าง |
| a | อะ | ia | เอียะ |
| aa | อา | iia | เอีย |
| I | อิ | va | เอือะ |
| ii | อี | vva | เอือ |
| v | อึ | ua | อัวะ |
| vv | อื | uua | อัว |
| u | อุ | 6 หน่วย | |
| uu | อู | | |
| e | เอะ | | |
| ee | เอ | | |
| x | แอะ | | |
| xx | แอ | | |
| o | โอะ | | |
| oo | โอ | | |
| @ | เอาะ | | |
| @@ | ออ | | |
| q | เออะ | | |
| qq | เออ | | |
| 18 หน่วย | | | |

| ตัวสะกด (Cf) | |
|---|---|
| เดี่ยว | ตัวอย่าง |
| p^ | พบ |
| t^ | เกร็ด |
| k^ | ปาก |
| n^ | หาร |
| m^ | ลม |
| ng^ | ฟาง |
| j^ | ยาย |
| w^ | กาว |
| เสียงทับศัพท์ | |
| f^ | กราฟ |
| l^ | แอล |
| s^ | เอส |
| ch^ | คลัช |
| 12 หน่วย | |

ที่มา: “Thai speech recognition tutorial [เว็บไซต์],” โดย หน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา, ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ

### 2.1.2 Robust Speech Recognition

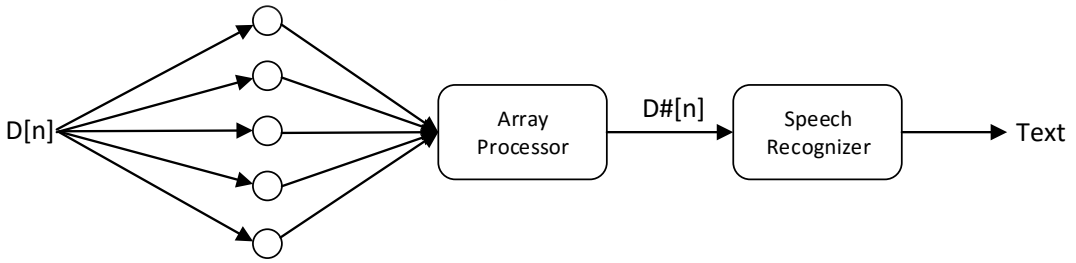
การรู้จำแบบทนทานต่อเสียงรบกวนนั้น มีหลายวิธีที่ใช้ในการทำให้การรู้จำนั้นทำงานได้ดีในสภาพแวดล้อมที่มีเสียงรบกวน เช่น งานวิจัยของ Seltzer (2003) นี้เป็นการวิเคราะห์เชิงความถี่ของเสียง โดยวิธีการลบเชิงสเปกตรัม อีกวิธีหนึ่งที่ทำให้การรู้จำทนทานต่อเสียงรบกวน คือ Microphone Array Processing

Microphone Array Processing เป็นการแก้ปัญหาเรื่องการรับสัญญาณเสียงที่ไม่ได้คุณภาพ เนื่องจากมีเสียงรบกวน และด้วยระยะห่างของ microphone กับผู้พูดที่มีผลต่อคุณภาพสัญญาณเสียง วิธีการนี้จะช่วยลดการบิดเบือนของสัญญาณเสียง และช่วยเพิ่มคุณภาพของสัญญาณ โดยการใช้ microphone หลายตัวในการรับสัญญาณ แทนที่จะเป็น microphone เพียงตัวเดียว สัญญาณที่ได้ก็จะไปผ่านกระบวนการ array processing ซึ่งเป็นการรวมสัญญาณที่ได้รับจากแต่ละ microphone นำมารวมกันเพื่อให้ได้สัญญาณเสียงที่ดี

โดยปกติแล้วการทำงานของระบบรู้จำที่ใช้ microphone arrays จะมีการแยกการทำงานเป็นอิสระต่อกันเป็น 2 ส่วน คือ ส่วนของการทำ array processing และ ส่วนของการรู้จำ ตามภาพที่ 2.8 ส่วนของ array processing เป็นส่วนที่จะพิจารณาในขั้นตอน pre-processing เพื่อที่จะทำให้คุณภาพของสัญญาณเสียงนั้นดีขึ้น ก่อนที่จะส่งไปทำการจำแนกคุณลักษณะสำคัญของเสียง และทำการรู้จำต่อไป

ภาพที่ 2.8

กระบวนการโดยทั่วไปของระบบรู้จำที่ใช้ microphone array โดยวัตถุประสงค์ของ array processing เพื่อที่จะได้สัญญาณเสียงที่มีคุณภาพดีที่สุด




ที่มา: “Microphone Array Processing for Robust Speech Recognition,” by Seltzer, M. L., 2003, p. 3

หลักการนำคลื่นเสียงหลักเพื่อมาใช้ใน microphone array processing นี้เรียกว่า beamforming ซึ่งโดยวิธีทั่วไปของ beamforming นี้เรียกว่า delay-and-sum ด้วยวิธีนี้สัญญาณที่ได้รับมาจากไมโครโฟนอาเรย์ จะมีการจัดเรียงสัญญาณเสียงของแต่ละไมโครโฟน เพื่อที่จะปรับในเรื่องของความยาวของสัญญาณเสียงที่แตกต่างกันระหว่างแหล่งที่มาของเสียงของไมโครโฟนแต่ละตัว ซึ่งก็มีการต่อยอดจากวิธี delay-and-sum beamforming ไปเป็นวิธี filter-and-sum beamforming ในแต่ละไมโครโฟนก็จะมีการเชื่อมต่อตัวกรองสัญญาณเสียง หลังจากผ่านตัวกรองสัญญาณแล้วก็จะนำมารวมกันเป็นสัญญาณเสียงเดียว

2.1.2.1 The LIMABEAM Algorithm

ปัญหาของ microphone array processing ในการทำการรู้จำนั้น เป็นเรื่องของการแยกประเภทของรูปแบบของสัญญาณเสียงที่ถูกแปลงไปเป็นลำดับของชุดค่าคุณลักษณะ (feature vectors) ที่จะต้องทำให้ได้กลุ่มของสัญญาณเสียงที่ถูกต้องมากที่สุด

อัลกอริทึม LIMABEAM นั้นทำขึ้นมาเพื่อเพิ่มประสิทธิภาพของไมโครโฟนอาเรย์ โดยวิธีการหาชุดของ array parameters ที่มีโอกาสสูงสุดที่จะได้เป็นกลุ่มของสัญญาณเสียงที่ถูกต้องแล้วใช้ข้อมูลจากการทำการรู้จำ นำกลับมาใช้ในการหา array parameters อีก ซึ่งเป็นการทำให้ประสิทธิภาพของการทำการรู้จำนั้นดีขึ้น

อัลกอริทึม LIMABEAM ของ Seltzer (2003) ได้มีการนำเอากระบวนการ filter-and-sum array processing มาใช้ในการช่วยชดเชยเสียงที่ผิดปกติจากเสียงรบกวน และ เสียงก้องกังวาน ได้อย่างดี โดยสามารถแสดงสูตรทางคณิตศาสตร์ได้ดังต่อไปนี้

$$x[k] = \sum_{m=1}^{M} h_m[k] * s_m[k] \tag{2.1}$$

โดยที่ $s_m[k]$ เป็นสัญญาณเสียงที่แยกตามช่วงเวลาของ microphone ตัวที่ $m$

$h_m[k]$ เป็นตัวกรองสัญญาณแบบ FIR ของ microphone ตัวที่ $m$

$x[k]$ เป็นผลลัพธ์ของ beamformer

$*$ แสดงถึง convolution

$k$ เป็น index ของเวลา

ค่าสัมประสิทธิของชุดตัวกรอง Finite Impulse Response (FIR) ทั้งหมดของ microphone ทุกตัวสามารถแสดงได้เป็นแบบ super-vector $h$ ในแต่ละ frame สามารถทำการ recognition features ได้จากฟังก์ชั่น $h$ ดังนี้

$$y_L(h) = \log_{10}(W \mid FFT(x(h))|^2) \tag{2.2}$$

$x(h)$ คือ Observation vector

$| FFT(x(h))|^2$ คือ vector ของ power spectrum components

$W$ คือ Mel filter matrix

$y_L(h)$ คือ vector ของ Log Filter Bank Energies (LFBE)

ค่าสัมประสิทธิ Cepstral ได้มาจากการผ่าน DCT rotation ดังนี้

$$y_C(h) = DCT(y_L(h)) \tag{2.3}$$

LIMABEAM นั้นได้มาจากชุดของตัวกรอง FIR ของแต่ละ microphone ซึ่งความเป็นไปได้มากที่สุดของ $y_C(h)$ จะให้ผลของ state sequence ของ hypothesized transcription ที่ถูกประเมินแล้ว ซึ่งแสดงได้ดังนี้

$$\hat{h} = \arg\max_h P(y_L(h)|w) \tag{2.4}$$

$w$ คือ hypothesized transcription

$P(y_L(h)|w)$ คือ ความน่าจะเป็นของ Observation features จะได้ transcription ที่ถูกพิจารณาแล้ว

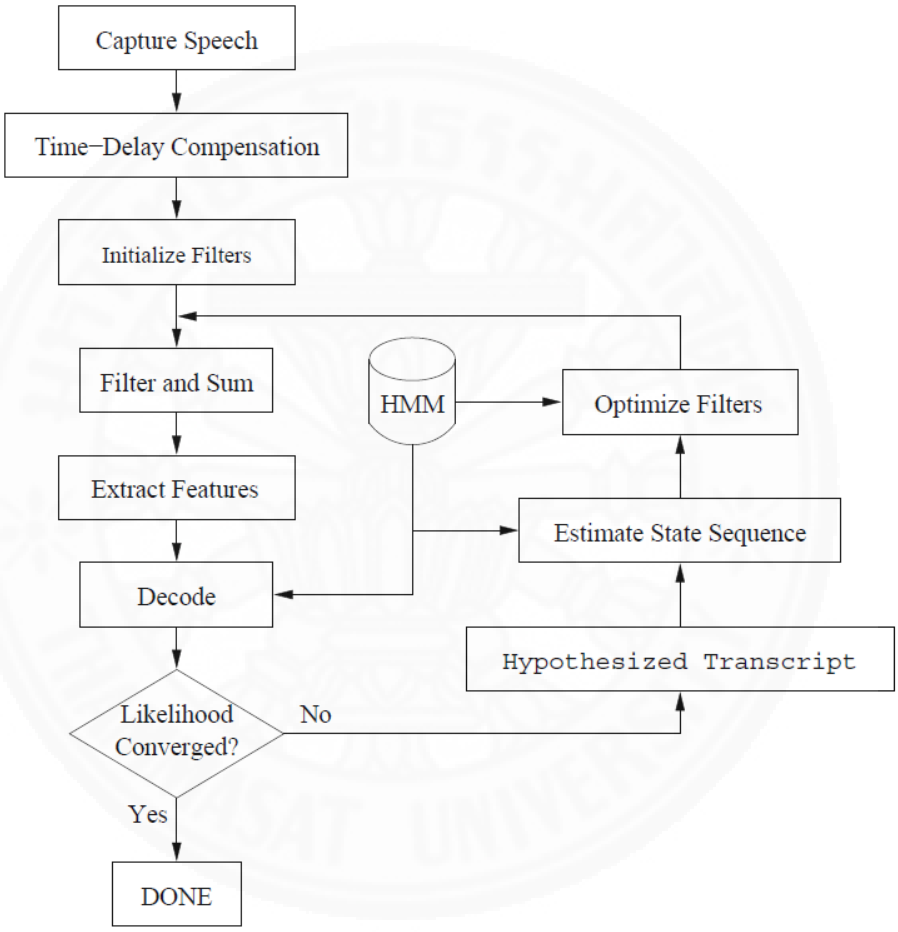
$\hat{h}$ คือ FIR parameter ของ super-vector

การทำ Optimization ถูกทำผ่าน non-linear Conjugate Gradient ส่วน state sequence สามารถประเมินได้จากผลลัพธ์ของ array beamformer (Unsupervised LIMABEAM)

Unsupervised LIMABEAM นั้นทำงานได้ดีในสภาพแวดล้อมที่มีเสียงรบกวน (noisy) ถึงแม้ว่าใช้เพียง 1 ช่องทางในการรับสัญญาณเสียงก็ตาม

ภาพที่ 2.9

แผนผังแสดงการทำงานของ Unsupervised LIMABEAM




ที่มา: “Microphone Array Processing for Robust Speech Recognition,” by Seltzer, M. L., 2003, p. 59

จากภาพที่ 2.9 เป็นแผนผังแสดงการทำงานของอัลกอริทึม Unsupervised LIMABEAM ซึ่งเริ่มจากการรับสัญญาณเสียงพูดเข้ามา แล้วทำการกระบวนการ Time Alignment โดยการหาค่าช่วงเวลาที่ delay ของแต่ละช่องรับสัญญาณ แล้วทำการเริ่มกำหนดค่า filter เพื่อทำการ filter and sum สัญญาณเสียงของทุกช่องรับสัญญาณเข้าด้วยกัน แล้วจึงนำสัญญาณเสียงที่

ได้มาทำการสกัดหาค่าคุณลักษณะสำคัญ (feature extraction) หลังจากนั้นเข้าสู่กระบวนการถอดรหัสสัญญาณ (decode) เพื่อทำการหาผลลัพธ์ที่ได้จากการรู้จำ หลังจากนั้นก็จะเช็คค่าความน่าจะเป็นของผลลัพธ์ว่า convergence หรือยัง ซึ่งถ้ายังก็จะนำผลลัพธ์ที่ได้ (hypothesized transcript) ไปทำการประเมิน state sequence แล้วทำการ optimize filter แล้วจึงกลับไปทำกระบวนการ filter and sum ใหม่วนไปทุกกระบวนการจนกว่าจะได้ค่าความน่าจะเป็นของผลลัพธ์ที่ convergence กันแล้วจึงถือว่าจบกระบวนการ และทำให้ได้ผลลัพธ์ที่ดีที่สุดออกมา

2.1.2.2 The N-best LIMABEAM Algorithm

LIMABEAM อัลกอริทึมนั้นเป็นการเพิ่มความน่าจะเป็นของการทำให้ได้ค่าสมมุติฐานของ transcription จากการทำขั้นตอนการรู้จำเพียงแค่ครั้งเดียว ส่วนในงานวิจัยนี้จะใช้วิธีการทำแบบ N-best เข้ามาประยุกต์ใช้ร่วมกับการรู้จำด้วยอัลกอริทึม LIMABEAM ซึ่งเป็นการเพิ่มประสิทธิภาพของการรู้จำของอัลกอริทึม LIMABEAM ให้ดียิ่งขึ้น โดยในขั้นตอนแรกจะทำการรู้จำให้ได้ค่าสมมุติฐานของ transcription ตามค่า N-best ที่กำหนด แล้วนำค่าสมมุติฐานของ transcription ที่ได้ไปทำการรู้จำต่อด้วยอัลกอริทึม LIMABEAM จนได้เป็นผลลัพธ์ของแต่ละชุด N-best แล้วทำการเลือกผลลัพธ์ที่ดีที่สุดจากชุด N-best ทุกตัว ก็จะได้เป็นผลลัพธ์สุดท้ายเป็นค่าสมมุติฐานของ transcription ของอัลกอริทึมนี้

ในขั้นตอนของการทำ N-best LIMABEAM ของ Brayda, Wellekens, & Omologo (2006) ในแต่ละชุด ข้อมูลชุดสัญญาณเสียงจะถูกจัดเรียงสัญญาณ แล้วไปผ่านขั้นตอนการ optimization แล้วนำชุดข้อมูลสัญญาณเสียงที่ได้ไปผ่านตัวกรองสัญญาณ นำค่าที่ได้ไปหาค่าคุณลักษณะสำคัญอันใหม่ แล้วค่อยส่งไปทำการรู้จำต่อก็จะได้ค่าสมมุติฐานของ transcription ที่ถูกเลือก ซึ่งสามารถแสดงได้ดังสูตรนี้

$$\hat{n} = \arg\max_{n} P(y_C(\hat{h}_n)|\widehat{w}_n) \qquad (2.5)$$

$\widehat{w}_n$ คือ transcription ที่ได้มาจากกระบวนการรู้จำในครั้งที่ 2

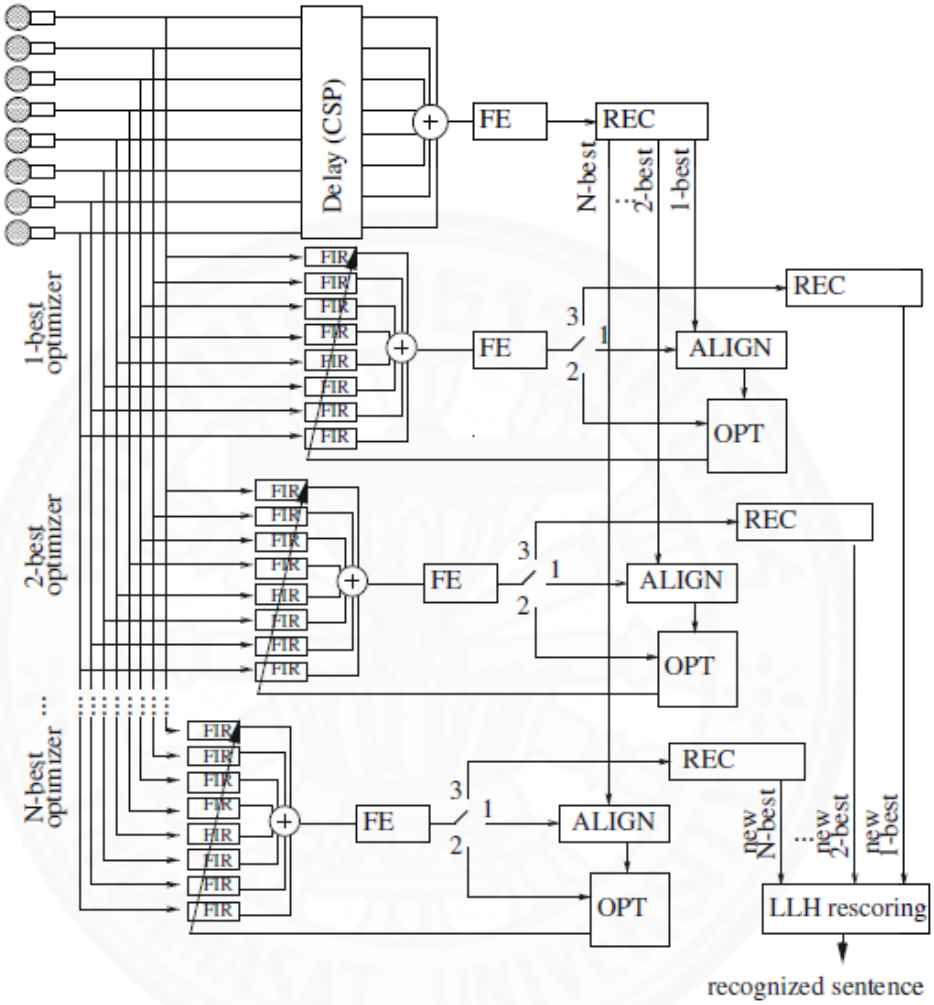
$\hat{n}$ คือ เป็น index ของ transcription ที่มีความน่าจะเป็นสูงสุด ซึ่งก็คือ $\widehat{w}_n$

การ optimization นั้นถูกทำเสร็จเรียบร้อยในขั้นตอน Log Filter Bank Energies (LFBE) domain ส่วนกระบวนการรู้จำนั้น ถูกทำในขั้นตอน Cepstral domain ค่าของความน่าจะเป็นได้ถูกปรับในขั้นตอน Cepstral domain เช่นเดียวกัน

ภาพรวมขั้นตอนการทำงานของอัลกอริทึม N-best LIMABEAM นั้นจะเริ่มจากการได้รับสัญญาณเสียงที่ได้มาจากกระบวนการของ microphone array โดยผ่านการรวมสัญญาณเสียงด้วยวิธี Delay and Sum (D&S) ซึ่งเป็นวิธีที่ใช้กันโดยทั่วไปในการรวมสัญญาณเสียงเข้าด้วยกัน หลังจากนั้นก็นำสัญญาณเสียงที่ได้มาทำการสกัดคุณลักษณะสำคัญของสัญญาณเสียง Feature Extraction (FE) แล้วนำคุณลักษณะสำคัญที่ได้ไปเข้าขั้นตอนการรู้จำ recognition (REC) ซึ่งในครั้งแรกจะได้ผลลัพธ์เป็น N-best hypotheses list ขึ้นมาตามการกำหนดค่า N-best หลังจากนั้นจะนำผลลัพธ์ hypothesis ของแต่ละ N-best ไปเข้ากระบวนการรู้จำอีกครั้งโดยใช้อัลกอริทึม LIMABEAM ซึ่งในขั้นตอนแรกก็จะทำการเลือก state sequence โดยใช้ Viterbi อัลกอริทึมหรือเรียกว่า Viterbi alignment (switch to 1: ALIGN) แล้วทำการแก้ไขปรับค่าพารามิเตอร์ หลังจากนั้นทำการ optimization ค่าสัมประสิทธิโดยใช้ Conjugate Gradient (switch to 2: OPT) แล้วนำสัญญาณเสียงจากชุด microphone array มาทำการรวมสัญญาเสียงใหม่โดยผ่านตัวกรอง FIR ซึ่งเรียกวิธีนี้ว่า Filter and Sum (F&S) แล้วนำสัญญาณเสียงที่ได้ไปทำการสกัดคุณลักษณะพิเศษ แล้วก็ทำการรู้จำ N-best feature ที่ได้มา (switch to 3: REC) นำผลลัพธ์ที่ได้มาเปรียบเทียบและทำการ alignment ซ้ำจนกว่าจะได้ค่าที่ convergence กันแล้วก็จะได้ผลลัพธ์ที่ดีที่สุดของ N-best แต่ละชุด และหลังจากนั้นก็ทำจนครบทุก N-best hypotheses list สุดท้ายก็จะนำเอาผลลัพธ์ที่ได้ในแต่ละชุด N-best มาทำการเปรียบเทียบค่าเพื่อหาค่าผลลัพธ์ที่ดีที่สุดเป็น new N-best Log-LikeliHoods (LLH-rescoring) โดยการเลือกค่าที่สูงที่สุด ก็จะได้ผลลัพธ์ที่ถูก recognized ออกมา สามารถแสดงได้ตามภาพที่ 2.10

ภาพที่ 2.10

Block diagram ของอัลกอริทึม N-best LIMABEAM




ที่มา: “Improving robustness of a likelihood-based beamformer in a real environment for automatic speech recognition, “ by Brayda, L., Wellekens, C., & Omologo, M.

## 2.2 งานวิจัยที่เกี่ยวข้อง

### 2.2.1 การพัฒนาระบบการรู้จำเสียงพูดภาษาไทย

ในส่วนของงานวิจัยที่เกี่ยวกับการพัฒนาระบบการรู้จำเสียงพูดภาษาไทย ได้มีการทำโดย บุญเสริม กิจศิริกุล และ ณัฐกร ทับทอง (2548) ซึ่งในงานวิจัยนี้เป็นการนำเอาเทคโนโลยี

ทางด้านการรู้จำเสียงพูดและการสังเคราะห์เสียงพูดภาษาไทยมาพัฒนาแล้วนำมาประยุกต์ใช้กับระบบสอบถามรายนามผู้ใช้โทรศัพท์แบบอัตโนมัติ โดยจำลองระบบลงบนเครื่องไมโครคอมพิวเตอร์รับเสียงข้อมูลขาเข้าโดยผ่านทางไมโครโฟน และส่งเสียงข้อมูลขาออกโดยผ่านทางลำโพง

ระบบสอบถามรายนามผู้ใช้โทรศัพท์แบบอัตโนมัตินั้นจะประกอบด้วย 2 ขั้นตอนใหญ่ ๆ ได้แก่ การรู้จำเสียงพูดภาษาไทย และการสังเคราะห์เสียงพูดภาษาไทย

ในขั้นตอนของการรู้จำเสียงพูด ซึ่งเป็นเสียงพูดของตัวเลขต่อเนื่องนั้น สามารถแบ่งเป็นขั้นตอนย่อยได้แก่ ขั้นตอนการหาจุดตัดหัวท้ายหน่วย และการตัดแบ่งพยางค์ ซึ่งผู้วิจัยได้เสนอวิธีในการตัดแบ่งโดยใช้ค่าพลังงานและค่าอัตราตัดผ่านแกนศูนย์ร่วมกันในการพิจารณาหาจุดแบ่ง ซึ่งให้ผลของอัตราความถูกต้องในการตัดแบ่งพยางค์เท่ากับ 86 เปอร์เซ็นต์ ส่วนขั้นตอนการรู้จำเสียงพูดได้ใช้คุณลักษณะสำคัญของเสียงคือ ราสต้า-พีแอลพี อันดับที่ 12 และอนุพันธ์อันดับที่ 1 และใช้โครงข่ายประสาทเทียมช่วยในการฝึกฝนและการรู้จำ ซึ่งได้ค่าพารามิเตอร์ที่เหมาะสมสำหรับการเรียนรู้จำเสียงพูดตัวเลขต่อเนื่องคือ จำนวนกรอบ 12 กรอบ จำนวนชั้นข้อมูลฮิดเดน 2 ชั้น จำนวนบัพของชั้นข้อมูลขาเข้า 288 บัพ จำนวนบัพของชั้นฮิดเดนที่หนึ่ง 50 บัพ จำนวนบัพของชั้นฮิดเดนที่สอง 100 บัพ จำนวนบัพของชั้นข้อมูลขาออก 10 บัพ ซึ่งได้ค่าอัตราความถูกต้องของระบบเท่ากับ 75 เปอร์เซ็นต์ และได้ค่าอัตราความถูกต้องระดับพยางค์ 94 เปอร์เซ็นต์

ส่วนในขั้นตอนการสังเคราะห์เสียงพูดใช้วิธีการสังเคราะห์เสียง โดยใช้วิธีการตัดคำโดยใช้เปรียบเทียบจากพจนานุกรม แล้วนำหน่วยเสียงที่ได้จากพจนานุกรมหน่วยเสียงมาต่อกันเป็นเสียงพูด ในงานวิจัยนี้ได้เพิ่มเติมส่วนที่เป็นคำอ่านของชื่อและนามสกุลเพื่อให้มีค่าความถูกต้องมากที่สุดคือ 100 เปอร์เซ็นต์ และได้เสนอวิธีการประมาณค่าความใกล้เคียงของคำพ้องเสียงในกลุ่มของพยัญชนะต้น และตัวสะกด โดยใช้ทฤษฎีเซตวิภัชนัยเข้ามาช่วยเพื่อลดจำนวนการเพิ่มคำศัพท์ในพจนานุกรม ซึ่งในชื่อและนามสกุลจะพบคำพ้องเสียงอยู่ 20 เปอร์เซ็นต์โดยประมาณ

### 2.2.2 วิธีการรู้จำเสียงพูดภาษาไทยแบบทนทานต่อเสียงรบกวนภายนอก

ในส่วนของงานวิจัยเกี่ยววิธีการรู้จำเสียงพูดภาษาไทยแบบทนทานต่อเสียงรบกวนภายนอก ได้มีการทำในงานวิจัยของ มนตรี โพธิโสโนทัย และ เฉลิมภัณฑ์ ฟองสมุทร (2554) ซึ่งได้

พูดถึงวิธีการรู้จำเสียงพูดภาษาไทยแบบทนทานต่อสัญญาณรบกวนจริงจากสภาพแวดล้อมภายนอก โดยวิธีการรู้จำเสียงนี้จะใช้ช่องสัญญาณจำนวนสองช่อง เพื่อใช้สำหรับรับเสียงพูดและใช้สำหรับรับเสียงรบกวน โดยเสียงรบกวนจะถูกกำหนดให้เป็นเสียงรบกวนที่เกิดขึ้น จากนั้นจะใช้วิธีการลบเชิงสเปกตรัม (Spectral subtraction: SS) เพื่อกำจัดเสียงรบกวนออก ขั้นตอนการทดลองได้ใช้เสียงพูดตัวเลขภาษาไทยจำนวน 10 คำ เป็นเสียงทดสอบ และได้เปรียบเทียบผลการทดลองกับสภาพแวดล้อมจริง โดยใช้ทดสอบกับสภาพแวดล้อมที่ไม่มีเสียงรบกวน และสภาพแวดล้อมที่มีเสียงรบกวนจริงที่ระดับความดังประมาณ 40–50 เดซิเบล ผลที่ได้พบว่าระบบที่ได้นำเสนอสามารถแยกเสียงตัวเลขได้ถูกต้องด้วยอัตราแยกเฉลี่ยประมาณ 98.0 เปอร์เซ็นต์ในกรณีที่ไม่มีเสียงรบกวน และอัตราแยกเฉลี่ยประมาณ 83.6 เปอร์เซ็นต์ในกรณีที่มีเสียงรบกวน

### 2.2.3 Microphone Array Processing

ในส่วนของงานวิจัยเกี่ยวกับ microphone array processing ได้มีการพูดถึงในงานวิจัยของ Seltzer (2003) ซึ่งพูดถึงหลักการนำคลื่นเสียงหลักมาใช้ใน microphone array processing ซึ่งเรียกว่า beamforming และวิธีที่ใช้กันทั่วไปที่สุด คือ delay-and-sum และมีการต่อยอดจาก delay-and-sum beamforming ไปเป็น filter-and-sum beamforming ซึ่งแต่ละ microphone มีการเชื่อมต่อตัวกรอง และสัญญาณที่ได้รับจะถูกกรองในครั้งแรก และนำมารวมกัน มีการทดลองในห้องที่มีเสียงรบกวนมีค่า SNR (6.5 dB) เป็นเสียงแอร์ เสียงพัดลมเครื่องคอมพิวเตอร์ และ เสียงรบกวนอื่น ๆ รวมทั้งมีเสียงสะท้อนในห้องที่วัดได้ 240 มิลลิวินาที ได้ค่าความถูกต้องของ delay-and-sum โดยใช้ 8 microphone อยู่ที่ 60% แต่ในการทดสอบนี้มีข้อจำกัดหากว่าผู้พูดมีการเคลื่อนที่ จะทำให้ค่า parameter ของ array processing มีการเปลี่ยนแปลงไป ส่งผลให้ความถูกต้องลดลง

### 2.2.4 The LIMABEAM Algorithm

ในส่วนของงานวิจัยเกี่ยวกับอัลกอริทึม LIMABEAM ได้มีการพูดถึงในงานวิจัยของ Seltzer (2003) ซึ่งทำขึ้นมาเพื่อเพิ่มประสิทธิภาพของ microphone array โดยวิธีการหาชุดของ array parameters ที่มีโอกาสสูงสุดที่จะได้ class เสียงที่ถูกต้องที่สุด โดยงานวิจัยนี้ได้พัฒนาอัลกอริทึมขึ้นมา 2 อัลกอริทึม คือ Calibrated LIMABEAM ซึ่งเป็นการเก็บข้อมูล transcription

ของผู้พูดไว้ก่อน เพื่อใช้ในการ optimize ค่า filter parameter ซึ่งวิธีนี้ก็จะเหมาะกับตำแหน่งของผู้พูดไม่มีการเคลื่อนที่ ส่วนอีกอัลกอริทึม คือ Unsupervised LIMABEAM ก็จะทำการ optimize parameter แยกอิสระโดยการใช้ hypothesized transcription จากการทำในครั้งแรก ซึ่งวิธีนี้ก็จะเหมาะกับตำแหน่งของผู้พูดที่มีการเคลื่อนที่ มีการทดลองในห้องที่มีเสียงรบกวนมีค่า SNR (6.5 dB) เป็นเสียงแอร์ เสียงพัดลมเครื่องคอมพิวเตอร์ และ เสียงรบกวนอื่น ๆ รวมทั้งมีเสียงสะท้อนในห้องในระดับปานกลางที่วัดได้ 0.24 วินาที ได้ค่าความถูกต้องของ Calibrated LIMABEAM อยู่ที่ 67.1% และค่าความถูกต้องของ Unsupervised LIMABEAM อยู่ที่ 69.8% ในการทดสอบนี้มีข้อจำกัดอยู่หากว่าเพิ่มค่าเสียงสะท้อนในห้องให้มากขึ้นก็จะทำให้ผลความถูกต้องลดลงมาก

### 2.2.5 The Subband-LIMABEAM Algorithm

ในส่วนของงานวิจัยเกี่ยวกับอัลกอริทึม Subband-LIMABEAM ได้มีการพูดถึงในงานวิจัยของ Seltzer (2003) ซึ่งทำขึ้นมาเพื่อเพิ่มประสิทธิภาพของอัลกอริทึม LIMABEAM ให้มีความทนทานต่อสภาพแวดล้อมที่มีเสียงสะท้อนอย่างมาก โดยการแบ่งสัญญาณออกมาเป็น subband แล้วดำเนินการตามกระบวนการตามปกติแบบ fullbrand มีการทดลองในห้องที่มีเสียงรบกวนเป็นเสียงแอร์ เสียงพัดลมเครื่องคอมพิวเตอร์ และ เสียงรบกวนอื่น ๆ รวมทั้งมีเสียงสะท้อนในห้องในระดับปานกลางที่วัดได้ 0.3 วินาที ได้ค่าความถูกต้องที่ดีขึ้นของ Calibrated S-LIMABEAM อยู่ที่ 36.2% และค่าความถูกต้องที่ดีขึ้นของ Unsupervised S-LIMABEAM เทียบกับ delay-and-sum อยู่ที่ 23.4% ในการทดสอบนี้มีข้อเสียอยู่ที่ต้องใช้ประสิทธิภาพในการประมวลผลสูง

### 2.2.6 The N-best LIMABEAM Algorithm

ในส่วนของงานวิจัยเกี่ยวกับอัลกอริทึม N-best LIMABEAM ได้มีการพูดถึงในงานวิจัยของ Brayda, Wellekens, & Omologo (2006) ซึ่งใช้วิธีการทำ N-best โดยการทำ optimization แบบแยกอิสระจากกัน และทำแบบคู่ขนานกันไป วิธีการนี้ตั้งอยู่บนพื้นฐานของความเป็นจริงที่ว่า N-best hypotheses list ก่อนการทำ optimization จะต้องถูกจัดลำดับโดยความน่าจะเป็น ซึ่งไม่จำเป็นต้องจัดลำดับตาม Word Error Rate (WER) โดยการใช้ LIMABEAM

อัลกอริทึมในแต่ละ hypothesis การจัดเรียงลำดับของ N-best list ก็จะมีการเปลี่ยนแปลงไป มีการทดลองในห้องที่มีเสียงรบกวนโดยการเปิด speaker 8 จุด ได้ค่าความถูกต้องอยู่ที่ 83.83%

จะเห็นได้ว่า งานวิจัยที่เกี่ยวข้องกับการรู้จำแบบทนทานต่อเสียงรบกวนสำหรับเสียงพูดภาษาไทยนั้นยังมีไม่มาก อีกทั้งการรู้จำแบบทนทานต่อเสียงรบกวน ยังไม่มีการใช้อัลกอริทึม N-best LIMABEAM เข้ามาช่วยในการเพิ่มประสิทธิภาพการรู้จำสำหรับเสียงพูดภาษาไทย ดังนั้นงานวิจัยนี้จะนำเสนอการรู้จำแบบทนทานต่อเสียงรบกวนสำหรับเสียงพูดภาษาไทย โดยใช้อัลกอริทึม N-best LIMABEAM

ตารางที่ 2.2
แสดงการเปรียบเทียบข้อเด่นข้อจำกัดของงานวิจัย

| งานวิจัย | หลักการ | ข้อเด่น | ข้อจำกัด |
|---|---|---|---|
| Sullivan (1996) | หลักการนำคลื่นเสียงหลักมาใช้ใน microphone array processing ซึ่งเรียกว่า beamforming | แก้ปัญหาเรื่องการรับสัญญาณเสียงที่ไม่ได้คุณภาพ เนื่องจากมีเสียงรบกวน การรับเสียงในสภาพแวดล้อมที่มีเสียงสะท้อน และระยะห่างของ microphone กับผู้พูด | หากผู้พูดมีการเคลื่อนที่จะทำให้ประสิทธิภาพในการรู้จำลดลง |
| Seltzer, M. L. (2003) | LIMABEAM โดยวิธีการหาชุดของ array parameters ที่มีโอกาสสูงสุดที่จะได้ class เสียงที่ถูกต้อง แล้วใช้ข้อมูลจากการทำ speech recognition นำกลับมาใช้ในการหา array parameters อีก | เพิ่มประสิทธิภาพของ microphone array processing ให้ดีขึ้น | ถ้ามีเสียงสะท้อนในห้องให้มากขึ้นก็จะทำให้ผลความถูกต้องลดลงมาก |

| | | | |
|---|---|---|---|
| Seltzer, M. L. (2003) | The Subband-LIMABEAM Algorithm โดยการแบ่งสัญญาณออกมาเป็น subband แล้วดำเนินการตามกระบวนการตามปกติแบบ fullbrand | เพิ่มประสิทธิภาพของอัลกอริทึม LIMABEAM ให้มีความทนทานต่อสภาพแวดล้อมที่มีเสียงสะท้อนอย่างมาก | ต้องใช้ประสิทธิภาพในการประมวลผลสูง |
| Brayda, L., Wellekens, C., & Omologo, M. (2006) | The N-best LIMABEAM Algorithm ทำ optimization parameters ตามจำนวนค่า N แบบแยกอิสระจากกัน | เพิ่มประสิทธิภาพของอัลกอริทึม LIMABEAM ให้ได้ค่าความถูกต้องที่ดีขึ้น | ยังทำงานได้ไม่ดีพอในสภาพแวดล้อมที่มีเสียงสะท้อนมาก |

จากตารางเปรียบเทียบข้อเด่นข้อจำกัดของงานวิจัยในตารางที่ 2.2 งานวิจัยที่เกี่ยวกับการรู้จำแบบทนทานต่อเสียงรบกวนที่อยู่ในสภาวะแวดล้อมจริง ที่มีการใช้หลักการของ microphone array เข้ามาใช้ในการรู้จำประโยคสนทนา ที่ผ่านมายังไม่มีการนำมาทดลองกับเสียงพูดภาษาไทยมาก่อน ดังนั้นในงานวิจัยนี้จึงเสนอการนำหลักการการของ microphone array ที่มีการรับสัญญาณเสียงจากหลายช่องทาง โดยการใช้อัลกอริทึม N-best LIMABEAM มาใช้สำหรับประโยคสนทนาที่เป็นภาษาไทย

# บทที่ 3

# วิธีดำเนินงานวิจัย

งานวิจัยนี้ได้ทำเกี่ยวกับ การรู้จำแบบทนทานต่อเสียงรบกวนสำหรับเสียงพูดภาษาไทย โดยใช้อัลกอริทึม N-best LIMABEAM เพื่อเป็นการเพิ่มประสิทธิภาพในการรู้จำให้มากขึ้น ในส่วนนี้จะกล่าวถึง เครื่องมือและซอฟต์แวร์ที่ใช้ในงานวิจัย ภาพรวมของระบบ ขั้นตอนการทดลอง และวิธีการวัดผลการทดลอง

## 3.1 เครื่องมือและซอฟต์แวร์ที่ใช้ในงานวิจัย

เครื่องมือและซอฟต์แวร์ที่ใช้ในงานวิจัยการรู้จำแบบทนทานต่อเสียงรบกวนสำหรับเสียงพูดภาษาไทย โดยใช้อัลกอริทึม N-best LIMABEAM มีรายละเอียดดังต่อไปนี้

ตารางที่ 3.1
แสดงรายละเอียดของเครื่องมือและซอฟต์แวร์ที่ใช้ในงานวิจัย

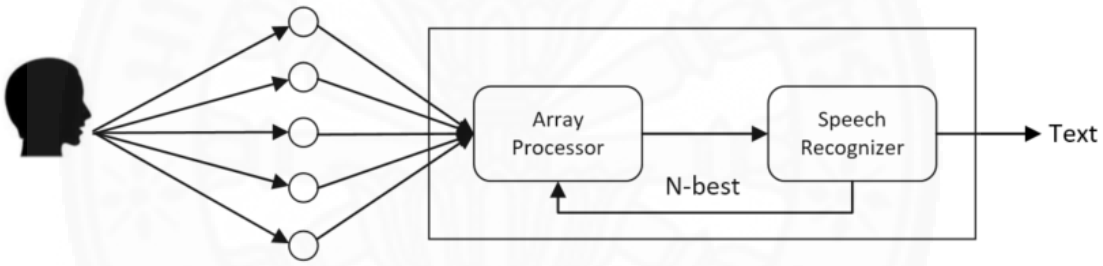
| | |
|---|---|
| ฮาร์ดแวร์ (Hardware) | **Notebook:**<br>Intel(R) Core(TM) i5-4300U CPU @ 1.90GHz<br>2.50GHz, 4.00 GB of RAM<br>**Smartphone:**<br>1. Samsung Galaxy Note III<br>2. Samsung Galaxy Note II<br>3. Samsung Galaxy Tab S |
| ระบบปฏิบัติการ (Operating System) | Microsoft Windows 8.1 Pro |

| | |
|---|---|
| ซอฟต์แวร์ (Software) | 1. Mobile App: Easy Voice Recorder v.1.9<br>2. Cool Edit Pro v.2.1<br>3. Notepad++ v.6.7.5<br>4. Hidden Markov Toolkit (HTK) v.3.4.1<br>5. MATLAB R2013a v.8.1<br>6. Microsoft Visual Studio 2010 |

## 3.2 ภาพรวมของระบบ

ภาพที่ 3.1

ภาพรวมของกระบวนการรู้จำด้วยอัลกอริทึม N-best LIMABEAM



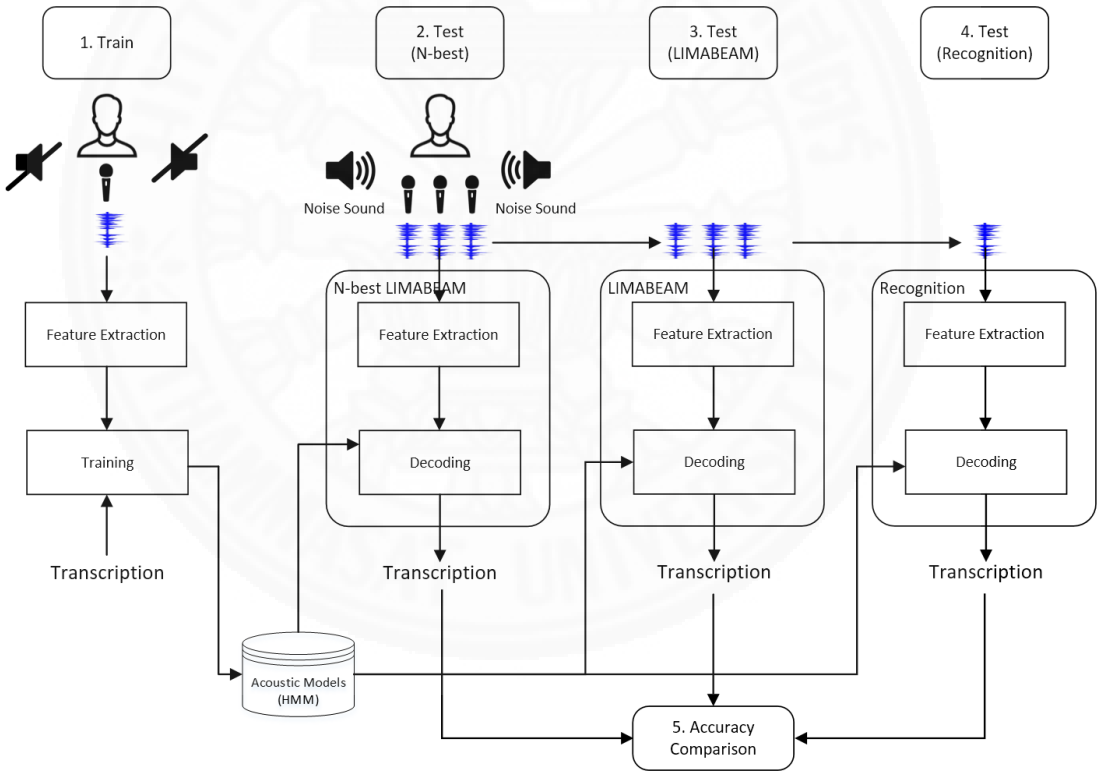

จากภาพที่ 3.1 จะเป็นภาพรวมของกระบวนการรู้จำเสียงพูดภาษาไทย โดยจะมีกระบวนการทำงานหลักอยู่ 2 ส่วน คือ หน่วยประมวลผลแบบอาเรย์ (Array Processor) และหน่วยการรู้จำเสียง (Speech Recognizer) หน่วยประมวลผลแบบอาเรย์จะรับสัญญาณเสียงพูดเข้ามาจากชุด microphone array หลายตัวเพื่อทำการรวมสัญญาณเสียงให้ได้สัญญาณเสียงที่มีคุณภาพ แล้วทำการส่งสัญญาณเสียงที่ผ่านกระบวนการแล้วไปให้ยังหน่วยการรู้จำเสียง เพื่อสกัดสัญญาณเสียงและทำกระบวนการรู้จำเสียง โดยกระบวนการรู้จำเสียงจะทำตามจำนวนรอบของการกำหนดค่า N-best ซึ่งการทำในแต่ละรอบจะมีการปรับค่าพารามิเตอร์ แล้วนำผลลัพธ์ที่ได้ในแต่ละรอบมาทำการเลือกเอาผลลัพธ์ที่มีค่าความแม่นยำที่สุดเป็นผลลัพธ์สุดท้าย

## 3.3 ขั้นตอนการทดลอง

### 3.3.1 ภาพรวมขั้นตอนการทดลอง

การทดลองนี้ได้ใช้ HTK HMM-based recognizer เป็นเครื่องมือในการรู้จำเสียง และใช้โปรแกรม MATLAB ในการประมวลผลเสียง ซึ่งภาพรวมของการทดลองมีขั้นตอนในการทดลองตามภาพที่ 3.2 โดยมีรายละเอียดดังนี้

ภาพที่ 3.2
ภาพรวมขั้นตอนการทดลอง




1. ขั้นแรกทำการฝึกฝน (Train) ข้อมูลเสียงเพื่อจัดทำเป็นฐานข้อมูลรูปแบบเสียง (Acoustic Models) ในแบบ HMM-based โดยทำการบันทึกเสียงด้วย microphone ตัวเดียว และอยู่ในสภาพแวดล้อมที่ไม่มีเสียงรบกวน เพื่อให้ได้เสียงที่สะอาด (Clean Signal) การบันทึกเสียงนี้จะใช้บทสนทนาการจองโรงแรมเป็นรูปแบบประโยคในการฝึกฝน ซึ่งมีจำนวนประโยคทั้งหมด 445 ประโยค ในการบันทึกเสียงนี้ใช้เสียงพูดของผู้ชายจำนวน 231 ประโยค เสียงพูดของผู้หญิงจำนวน

214 ประโยค มีจำนวนคำทั้งหมด 4,069 คำ เป็นคำที่ไม่ซ้ำกันทั้งหมด 210 คำ ดังตัวอย่างประโยคในตารางที่ 3.2 หลังจากทำการบันทึกเสียงเสร็จแล้ว ก็นำไฟล์เสียงที่ได้ไปทำการสกัดหาค่าคุณลักษณะสำคัญ (Feature Extraction) เมื่อได้ค่าคุณลักษณะสำคัญแล้ว ก็นำไปทำการ Training โดยใช้ข้อมูลบทสทนา (Transcription) ร่วมด้วย สุดท้ายก็จะได้เป็นฐานข้อมูลรูปแบบเสียงเพื่อเตรียมเอาไว้ใช้ในขั้นตอนการทดสอบต่อไป

ตารางที่ 3.2
ตัวอย่างประโยคที่ใช้ในการฝึกฝน

| ลำดับ | เพศผู้พูด | ประโยค |
|---|---|---|
| 1 | ชาย | สอง คน ครับ ผม เข้า เสาร์ ที่ สิบ สี่ ธันวา ออก เช้า วัน ต่อไป ครับ |
| 2 | ชาย | สอง คน ครับ เช็คอิน วันนี้ และ เช็คเอาท์ พรุ่งนี้ เช้า ครับ |
| 3 | ชาย | ผม มา พัก คน เดียว ครับ |
| 4 | ชาย | จำนวน สอง ท่าน ครับ พัก วันนี้ เช็คเอาท์ พรุ่งนี้ ครับ |
| 5 | ชาย | สอง คน ตั้งแต่ วัน ที่ สิบ สาม ถึง สิบ สี่ ธันวา ครับ |
| 6 | หญิง | สอง คน ค่ะ พัก คืน นี้ คืน เดียว เช็คเอาท์ พรุ่งนี้ ค่ะ |
| 7 | หญิง | มา สอง คน ค่ะ พัก คืน นี้ ออก พรุ่งนี้ ประมาณ เที่ยง เที่ยง ค่ะ |
| 8 | หญิง | สอง คน ค่ะ เข้า พัก วัน ที่ ยี่สิบ ออก วัน ที่ ยี่สิบ เอ็ด ค่ะ |
| 9 | หญิง | พัก สอง คน ขอ ห้อง รวม เข้า วันนี้ ออก พรุ่งนี้ เที่ยง ค่ะ |
| 10 | หญิง | สอง คน วันนี้ จะ เข้า พัก แล้ว คงจะ เช็คเอาท์ พรุ่งนี้ เลย คะ |

2. ขั้นทำการทดสอบ (Test) โดยใช้อัลกอริทึม N-best LIMABEAM ขั้นตอนนี้จะทำการบันทึกเสียงในสภาพแวดล้อมที่มีเสียงรบกวน เพื่อนำสัญญาณเสียงที่ได้ไปผ่านกระบวนการรู้จำเสียงโดยใช้อัลกอริทึม N-best LIMABEAM เริ่มจากการบันทึกเสียงโดยใช้ microphone array และอยู่ในสภาพแวดล้อมที่มีเสียงรบกวนรอบข้าง โดยทำการบันทึกเสียงสำหรับทดสอบจำนวน 10 ประโยค ดังแสดงในตารางที่ 3.3 แล้วนำไฟล์เสียงไปเข้ากระบวนการรู้จำเสียงโดยใช้อัลกอริทึม N-best LIMABEAM ซึ่งก็จะมีการสกัดหาค่าคุณลักษณะสำคัญ (Feature Extraction) และการถอดค่าคุณลักษณะสำคัญ (Decoding) โดยใช้ฐานข้อมูลรูปแบบเสียงที่ได้จัดทำไว้แล้วในขั้นตอนการ

ฝึกฝน เป็นข้อมูลตั้งต้นในการรู้จำเสียง จนสุดท้ายก็จะได้ผลลัพธ์เป็นข้อมูลบทสทนา (Transcription) ออกมาพร้อมค่าความถูกต้องแม่นยำ (%Correct and %Accuracy)

ตารางที่ 3.3
ประโยคที่ใช้ในการทดสอบ

| ลำดับ | ประโยค |
|---|---|
| 1 | ผม ต้องการ ห้องพัก ซัก ห้อง จะ เข้า พัก วัน ที่ สิบ ห้า และ ออก สิบ หก พฤศจิกา ครับ |
| 2 | สอง คน ครับ เข้า วัน ที่ ยี่สิบ สาม เดือน นี้ แล้ว ก็ เช็คเอาท์ ออก วันรุ่งขึ้น ครับ |
| 3 | เอ่อ ผม กับ เพื่อน จำนวน ทั้งหมด สอง คน ต้องการ พัก หนึ่ง คืน ครับ |
| 4 | สวัสดี ครับ ผม มา กัน ทั้งหมด สอง คน ครับ พรุ่งนี้ ถึง จะ กลับ ครับ |
| 5 | ผม มา กับ เพื่อน อีก คน จะ เข้า พัก วันนี้ และ เช็คเอาท์ พรุ่งนี้ ครับ |
| 6 | สาม คน ครับ เข้า พัก คืน นี้ และ พรุ่งนี้ จะ เช็คเอาท์ ออก ก่อนเที่ยง ครับ |
| 7 | ผม ต้องการ จอง ห้องพัก ครับ ผม จะ เข้า พัก กับ เพื่อน อีก คน หนึ่ง และ จะ พัก กัน หนึ่ง คืน |
| 8 | สวัสดี ครับ ผม กับ เพื่อน อีก หนึ่ง คน ครับ คิด ว่า จะ พัก ซัก คืน พรุ่งนี้ ก็ จะ เดินทาง ต่อ ครับ |
| 9 | อืม สอง ท่าน ครับ เข้า พัก บ่าย เสาร์ ที่ สิบ สี่ และ ออก เที่ยง ที่ สิบ ห้า เลย ครับ |
| 10 | เข้า พัก สอง คน ครับ แล้ว ก็ จะ อยู่ แค่ คืน เดียว ครับ คงจะ เช็คเอาท์ พรุ่งนี้ ประมาณ สิบ โมง เช้า |

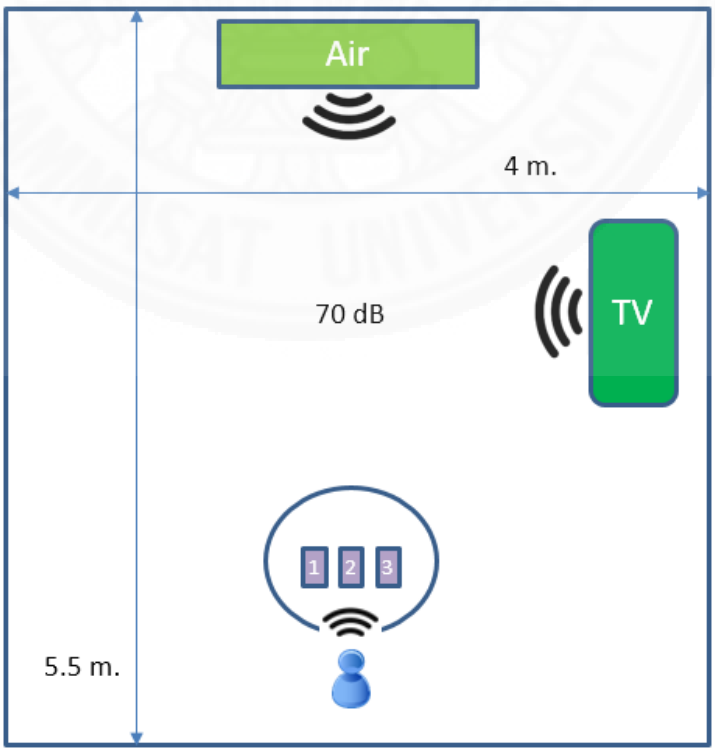
3. ขั้นทำการทดสอบ (Test) โดยใช้อัลกอริทึม LIMABEAM ในขั้นตอนนี้จะนำไฟล์เสียงที่ได้จากการบันทึกเสียงในขั้นตอนการทดสอบโดยใช้อัลกอริทึม N-best LIMABEAM นำไปผ่านกระบวนการรู้จำเสียงโดยใช้อัลกอริทึม LIMABEAM โดยเริ่มจากการสกัดหาค่าคุณลักษณะสำคัญ (Feature Extraction) และการถอดค่าคุณลักษณะสำคัญ (Decoding) โดยใช้ฐานข้อมูลรูปแบบเสียงที่ได้จัดทำไว้แล้วในขั้นตอนการฝึกฝน เป็นข้อมูลตั้งต้นในการรู้จำเสียง จนสุดท้ายก็จะได้ผลลัพธ์เป็นข้อมูลบทสทนา (Transcription) ออกมาพร้อมค่าความถูกต้องแม่นยำ (%Correct and %Accuracy)

4. ขั้นทำการทดสอบ (Test) โดยใช้วิธีการรู้จำแบบปกติทั่วไป (Recognition) ในขั้นตอนนี้จะนำไฟล์เสียงที่ได้จากการบันทึกเสียงในขั้นตอนการทดสอบโดยใช้อัลกอริทึม N-best LIMABEAM แต่นำเพียงไฟล์เสียงจาก microphone เพียงตัวเดียว นำไปผ่านกระบวนการรู้จำเสียง โดยเริ่มจากการสกัดหาค่าคุณลักษณะสำคัญ (Feature Extraction) และการถอดค่าคุณลักษณะสำคัญ (Decoding) โดยใช้ฐานข้อมูลรูปแบบเสียงที่ได้จัดทำไว้แล้วในขั้นตอนการฝึกฝน เป็นข้อมูลตั้งต้นในการรู้จำเสียง จนสุดท้ายก็จะได้ผลลัพธ์เป็นข้อมูลบทสทนา (Transcription) ออกมาพร้อมค่าความถูกต้องแม่นยำ (%Correct and %Accuracy)

5. ขั้นการเปรียบเทียบค่าความแม่นยำ (Accuracy Comparison) ในขั้นนี้ก็จะนำผลลัพธ์ที่ได้จากการทดสอบด้วยอัลกอริทึม N-best LIMABEAM, LIMABEAM และ Recognition มาทำการเปรียบเทียบค่าความแม่นยำของผลลัพธ์ที่ได้ ว่าวิธีไหนจะมีความแม่นยำมากกว่ากัน

### 3.3.2 **สภาพแวดล้อมที่ใช้ในการทดลอง**

ภาพที่ 3.3
ห้องบันทึกเสียงการทดลอง



สภาพแวดล้อมที่ใช้ในการทดลองสำหรับการบันทึกเสียงพูดทั้งเสียงพูดแบบสะอาดและเสียงพูดแบบมีเสียงรบกวน เป็นห้องขนาด 5.5 x 4 เมตร ในห้องมีเสียงรบกวนเป็นเสียงเครื่องปรับอากาศ และ เสียงโทรทัศน์ โดยวัดค่าความดังได้ 70 dB ณ ตำแหน่งบันทึกเสียงวัดค่าคุณภาพสัญญาณเสียงต่อเสียงรบกวน (SNR) ได้ 1.7 dB และวัดค่าเวลาการสะท้อนกลับของเสียง (RT) ได้ 0.25s ทำการบันทึกเสียงโดยใช้โทรศัพท์ Smartphone จำนวน 3 เครื่อง ประกอบด้วย Samsung Galaxy Note II, Samsung Galaxy Note III และ Samsung Galaxy Tab S ด้วยค่า sample rate 16 kHz และค่า bit-depth 16 bit วางเรียงห่างกัน 15 เซนติเมตร ระยะห่างระหว่างผู้พูดกับโทรศัพท์ 30 เซนติเมตร เนื่องด้วยระยะห่างระหว่างผู้พูดกับโทรศัพท์เป็นระยะห่างของการพูดผ่านไมโครโฟนที่ตั้งอยู่บนโต๊ะแบบปกติทั่วไป จึงทำให้ต้องจำกัดจำนวนไมโครโฟนให้ได้ตามขนาดของพื้นที่ตั้งบนโต๊ะ อีกทั้งด้วยขนาดของโทรศัพท์ที่มีขนาดใหญ่ ตามภาพที่ 3.3

### 3.4 การวัดผลการทดลอง

ในการวัดผลการทดลองของงานวิจัยนี้ จะใช้การวัดประสิทธิภาพด้วยวิธีหาค่าความแม่นยำ (Accuracy) และค่าความถูกต้อง (Correct) ของชุดคำที่เป็นผลลัพธ์จากการรู้จำ วิธีการวัดประสิทธิภาพจะนำชุดคำที่เป็นผลลัพธ์จากการรู้จำ นำมาเปรียบเทียบกับชุดคำที่ถูกต้อง โดยใช้อัลกอริทึม Dynamic Programming ในการเปรียบเทียบ 2 ชุดคำ เพื่อทำการนับจำนวนคำที่จับคู่ตรงกันตามลำดับ (M: Match) และนับจำนวนคำที่ผิดพลาด ซึ่งประกอบด้วย คำที่ถูกเปลี่ยน (S: Substitution) คำที่ถูกลบ (D: Deletion) และคำที่เพิ่ม (I: Insertion) โดยแสดงเป็นสูตรการหาค่าความถูกต้องดังนี้

$$Percent\ Correct = \frac{\mathrm{N} - S - \mathrm{D}}{\mathrm{N}}\ x\ 100\% \tag{3.1}$$

$\mathrm{N}$ คือ จำนวนคำทั้งหมดในชุดคำที่ถูกต้อง

สูตรการหาค่าความแม่นยำดังนี้

$$Percent\ Accuracy = \frac{\mathrm{N} - S - \mathrm{D} - \mathrm{I}}{\mathrm{N}}\ x\ 100\% \tag{3.2}$$

ตัวอย่างการวัดค่าความแม่นยำ และค่าความถูกต้องจากผลลัพธ์การรู้จำหนึ่งประโยค ดังตารางที่ 3.4

ตารางที่ 3.4
ตัวอย่างผลลัพธ์การรู้จำของหนึ่งประโยค

| | |
|---|---|
| Transcript File: | trans068 |
| HTK En Result: | w116 w016 w075 w005 w174 w146 w126 w015 w146 w016 w018 w114 w028 w086 w035 w019 w079 w028 w028 w159 w140 w016 |
| HTK Th Result: | สวัสดี ครับ ผม กับ เพื่อน อีก หนึ่ง คน อีก ครับ คิด ว่าง จะ พัก ซัก คืน พรุ่งนี้ จะ จะ เดินทาง ออก ครับ |
| Master Transcript: | สวัสดี ครับ ผม กับ เพื่อน อีก หนึ่ง คน ครับ คิด ว่า จะ พัก ซัก คืน พรุ่งนี้ ก็ จะ เดินทาง ต่อ ครับ |
| WORD COMPARED: | MMMMMMMMIMMSMMMMMSMMSM |
| WORD: | %Correct=85.71, %Accuracy=80.95 [M=18, D=0, S=3, I=1, N=21] |

จากตัวอย่างค่า N=21 ค่า S=3 ค่า D=0 ค่า I=1 และค่า M=18

$$Percent\ Correct = \frac{21 - 3 - 0}{21}\ x\ 100 = 85.71\%$$

$$Percent\ Accuracy = \frac{21 - 3 - 0 - 1}{21}\ x\ 100 = 80.95\%$$

# บทที่ 4

# ผลการวิจัยและอภิปรายผล

## 4.1 การทดลอง

การทดลองนี้ได้ใช้ HTK HMM-based recognizer เป็นเครื่องมือในการรู้จำเสียง และใช้โปรแกรม MATLAB ในการประมวลผลเสียง ใช้บทสนทนาการจองโรงแรมเป็นบทสนทนาในการทดสอบ และจัดทำเป็นฐานข้อมูลเสียง มีประโยคที่ใช้ในการฝึกฝนการรู้จำ จำนวน 445 ประโยค แบ่งเป็นเสียงผู้ชายจำนวน 231 ประโยค เสียงผู้หญิงจำนวน 214 ประโยค จำนวนคำทั้งหมด 4,069 คำ เป็นคำที่ไม่ซ้ำกันทั้งหมด 210 คำ ทำการบันทึกเสียงในห้องตามภาพที่ 3.11

การทดลองเริ่มจากการบันทึกเสียงเพื่อการฝึกฝน (Training) โดยบันทึกเสียงในห้องที่ไม่มีเสียงรบกวนผ่าน Smartphone จำนวน 1 เครื่อง ต่อมาทำการบันทึกเสียงเพื่อการทดสอบ โดยบันทึกเสียงในห้องที่มีเสียงรบกวน และใช้เสียงของผู้พูดคนเดียวกับที่ใช้เสียงในการฝึกฝน บันทึกเสียงจำนวน 10 ประโยค โดยพูดผ่าน Smartphone จำนวน 3 เครื่อง แล้วนำชุดเสียงที่ได้ไปทำการทดลองเพื่อเปรียบเทียบประสิทธิภาพในรูปแบบต่าง ๆ โดยสรุปรูปแบบการทดลองได้ 4 รูปแบบดังต่อไปนี้

1. เปรียบเทียบผลการทดลองตามอัลกอริทึม LIMABEAM, N-best LIMABEAM และ เสียงรบกวนแบบไม่มีอัลกอริทึม
2. เปรียบเทียบการทดลองตามค่า N-best
3. เปรียบเทียบการทดลองตามค่าคุณภาพสัญญาณเสียงต่อเสียงรบกวน (SNR)
4. เปรียบเทียบการทดลองตามประเภทช่องรับสัญญาณ

โดยการเปรียบเทียบผลการทดลองตามอัลกอริทึม LIMABEAM, N-best LIMABEAM และ เสียงรบกวนแบบไม่มีอัลกอริทึม จะใช้ผลการรู้จำประโยค 10 ประโยค มาเปรียบเทียบค่าความถูกต้องในแต่ละวิธี ซึ่งจะแสดงให้เห็นถึงค่าการพัฒนาเชิงสัมพันธ์ที่ดีขึ้นในแต่ละอัลกอริทึมเมื่อเทียบกับเสียงรบกวนแบบไม่มีอัลกอริทึม

การเปรียบเทียบการทดลองตามค่า N-best นั้นจะทำการเปรียบเทียบค่าความถูกต้องตามค่า N-best ซึ่งกำหนดไว้ให้มีค่าเป็น 3, 5, 10 และ 15 โดยใช้จำนวนประโยค 10

ประโยคในการทดลอง และจะมีการเปรียบเทียบแสดงให้เห็นถึงเวลาที่ใช้ในการประมวลผลที่มีความสัมพันธ์กับค่า N-best

การเปรียบเทียบการทดลองตามค่าคุณภาพสัญญาณเสียงต่อเสียงรบกวน (SNR) จะทำการเปรียบเทียบค่า SNR ในแต่ละค่าเพื่อดูว่าค่า SNR ให้ค่าความถูกต้องที่แตกต่างกันอย่างไร โดยมีค่า SNR ทั้งหมด 6 ค่าด้วยกัน คือ 1.1 dB, 1.3 dB, 1.5 dB, 1.9 dB, 2.5 dB และ 3.4 dB

การเปรียบเทียบการทดลองตามประเภทช่องรับสัญญาณ โดยจะทำการเปรียบเทียบค่าความถูกต้องที่ได้ตามประเภทช่องรับสัญญาณเสียงแบบช่องทางเดียว (Single) กับช่องรับสัญญาณเสียงแบบหลายช่องทาง (Multiple) โดยใช้อัลกอริทึม LIMABEAM และ N-best LIMABEAM ในการเปรียบเทียบ ซึ่งจะใช้ข้อมูลเสียงแบบไม่มีเสียงรบกวน (Clean) และข้อมูลที่มีเสียงรบกวน (Noise) ในการทดสอบ

## 4.2 ผลการทดลอง

ในงานวิจัยนี้ได้กำหนดรูปแบบการทดลองไว้ 4 รูปแบบด้วยกัน โดยผลการทดลองในรูปแบบแรก คือ การเปรียบเทียบผลการทดลองตามอัลกอริทึม LIMABEAM, N-best LIMABEAM และ เสียงรบกวนแบบไม่มีอัลกอริทึม (No Algorithm) โดยใช้จำนวนประโยค 10 ประโยคในการทดลอง สรุปผลการทดลองได้ดังตารางที่ 4.1 และ ภาพที่ 4.1 โดยจากผลการทดลองอัลกอริทึม N-best LIMABEAM นั้นมีค่าความถูกต้องมากกว่าอัลกอริทึม LIMABEAM ยกเว้นประโยคที่ 9 ที่มีค่าน้อยกว่า ส่วนค่าความถูกต้องเมื่อเทียบกับ เสียงรบกวนแบบไม่มีอัลกอริทึม จะมีอยู่ 2 ประโยคที่อัลกอริทึม N-best LIMABEAM มีค่าน้อยกว่า คือ ประโยคที่ 6 และ 9 ซึ่งจะเห็นว่าผลการรู้จำในรายประโยคด้วยอัลกอริทึม N-best LIMABEAM ก็ยังมีอยู่บางประโยคที่ให้ค่าความถูกต้องน้อยกว่าอัลกอริทึม LIMABEAM และ กับเสียงรบกวนแบบไม่มีอัลกอริทึม

จากตารางที่ 4.2 เป็นการแสดงถึงการเปรียบเทียบค่าการพัฒนาเชิงสัมพันธ์ที่ดีขึ้นในแต่ละอัลกอริทึม เมื่อเทียบกับเสียงรบกวนแบบไม่มีอัลกอริทึม โดยภาพรวมของค่าความถูกต้องของทั้ง 10 ประโยคในแต่ละอัลกอริทึมนั้น อัลกอริทึม N-best LIMABEAM จะให้ค่าความถูกต้องดีที่สุดที่ 27.22% ซึ่งมีค่าการพัฒนาเชิงสัมพันธ์อยู่ที่ 19.61% โดยใช้ค่า N-best อยู่ที่ 10 ส่วนอัลกอริทึม LIMABEAM ให้ค่าความถูกต้องที่ 20.12% ซึ่งมีค่าการพัฒนาเชิงสัมพันธ์อยู่ที่ 11.76% ส่วนเสียงรบกวนแบบไม่มีอัลกอริทึมนั้นให้ค่าความถูกต้องที่ 9.47%

ตารางที่ 4.1

เปรียบเทียบค่าความถูกต้องการรู้จำตามอัลกอริทึมในแต่ละวิธีจำนวน 10 ประโยค

| ประโยค | No Algorithm (Noise) | LIMABEAM (Noise) | N-best (Noise) |
|---|---|---|---|
| 1 | 5.56 | 38.89 | 44.44 |
| 2 | -6.25 | -6.25 | -6.25 |
| 3 | -7.69 | 7.69 | 7.69 |
| 4 | 0 | -7.14 | 0 |
| 5 | 21.43 | 7.14 | 28.57 |
| 6 | 14.29 | 0 | 7.14 |
| 7 | 10.00 | 50.00 | 65.00 |
| 8 | 14.29 | 28.57 | 38.10 |
| 9 | 31.58 | 31.58 | 26.32 |
| 10 | 5.00 | 25.00 | 35.00 |

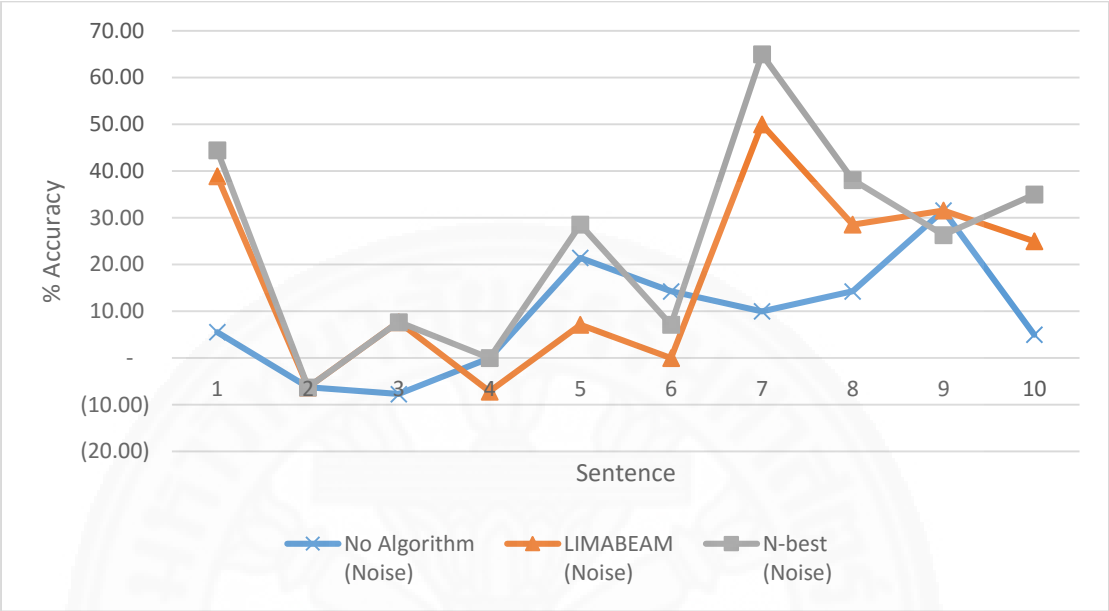
ตารางที่ 4.2

เปรียบเทียบค่าการพัฒนาเชิงสัมพันธ์ที่ดีขึ้นในแต่ละอัลกอริทึม

| วิธีการ | ความถูกต้อง | การพัฒนาเชิงสัมพันธ์ |
|---|---|---|
| No Algorithm | 9.47% | - |
| LIMABEAM | 20.12% | 11.76% |
| N-best LIMABEAM | 27.22% | 19.61% |

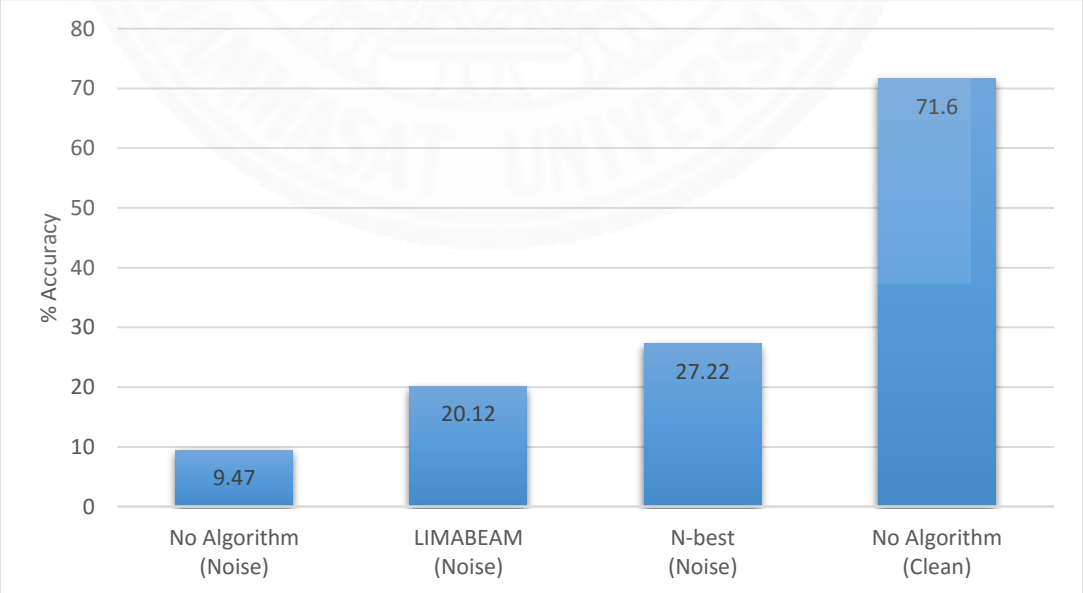
ภาพที่ 4.1

กราฟแสดงการเปรียบเทียบผลการรู้จำตามอัลกอริทึมในแต่ละวิธีจำนวน 10 ประโยค




ภาพที่ 4.2

กราฟภาพรวมแสดงการเปรียบเทียบผลการรู้จำในแต่ละวิธี



จากการทดลองในส่วนนี้ ได้ทำการเปรียบเทียบให้เห็นว่า ถึงแม้เสียงพูดที่ได้รับเข้ามามีคุณภาพเสียงที่ไม่ค่อยดีนัก หรือเรียกว่าเสียงนั้นมีเสียงรบกวนอยู่รอบข้าง แต่เมื่อนำไปผ่านอัลกอริทึม N-best LIMABEAM ก็ยังสามารถให้ค่าความถูกต้องแม่นยำได้ดีกว่าเสียงที่ไม่ผ่านอัลกอริทึมเลย หรือแม้กระทั่งเทียบกับอัลกอริทึม LIMABEAM เองก็ตาม ก็ยังให้ค่าที่ดีกว่า ซึ่งจากภาพที่ 4.2 เป็นการแสดงกราฟเปรียบเทียบผลการรู้จำในแต่ละวิธี จะเห็นว่าค่าของเสียงรบกวนแบบไม่มีอัลกอริทึม (Noise) นั้นให้ค่าที่ต่ำที่สุดคือ 9.47% ในขณะที่อัลกอริทึม LIMABEAM ให้ค่าที่ 20.12% และ N-best ให้ค่าที่ 27.22% แสดงว่าถ้าเสียงที่มี Noise ไม่ผ่านอัลกอริทึมใด ๆ เลย ค่าความถูกต้องแม่นยำจะต่ำมาก ส่วนเสียงที่ไม่มีเสียงรบกวนนั้น ให้ค่าความถูกต้องแม่นยำสูงที่สุดที่ 71.6% ซึ่งแน่นอนว่าค่าต้องได้สูงสุด เพราะเสียงที่นำไปเข้ากระบวนการรู้จำนี้เป็นเสียงที่อยู่ในสภาพแวดล้อมที่ไม่มีเสียงรบกวน และเป็นเสียงชุดเดียวกับที่ใช้ในการฝึกฝนฐานข้อมูล HMM จึงทำให้ได้ค่าที่สูงสุด

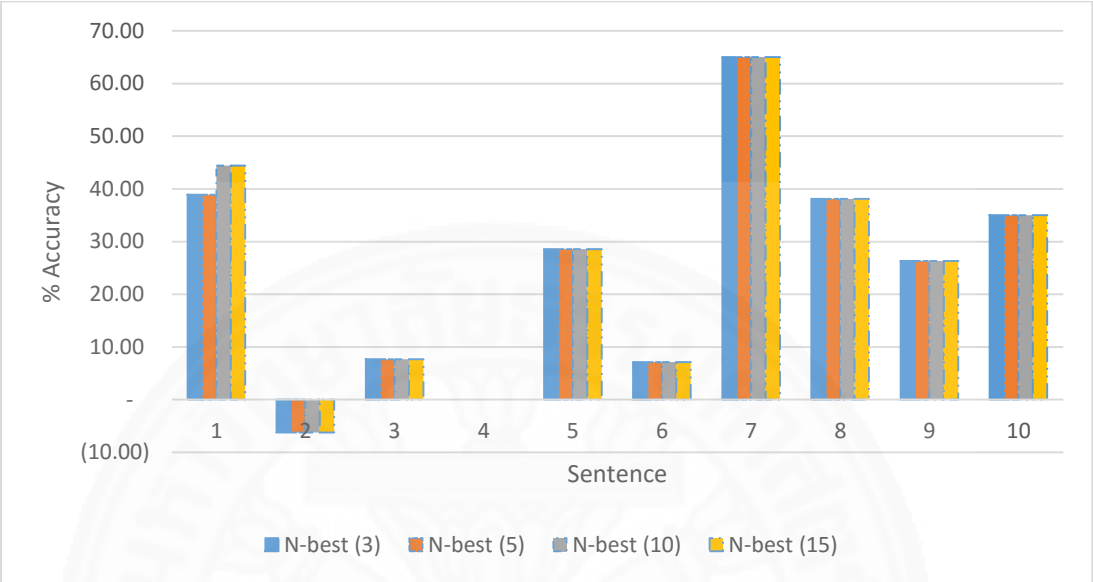
ตารางที่ 4.3

เปรียบเทียบผลการรู้จำตามค่า N-best จำนวน 10 ประโยค

| ประโยค | N-best (3) | N-best (5) | N-best (10) | N-best (15) |
|---|---|---|---|---|
| 1 | 38.89 | 38.89 | 44.44 | 44.44 |
| 2 | -6.25 | -6.25 | -6.25 | -6.25 |
| 3 | 7.69 | 7.69 | 7.69 | 7.69 |
| 4 | 0 | 0 | 0 | 0 |
| 5 | 28.57 | 28.57 | 28.57 | 28.57 |
| 6 | 7.14 | 7.14 | 7.14 | 7.14 |
| 7 | 65.00 | 65.00 | 65.00 | 65.00 |
| 8 | 38.10 | 38.10 | 38.10 | 38.10 |
| 9 | 26.32 | 26.32 | 26.32 | 26.32 |
| 10 | 35.00 | 35.00 | 35.00 | 35.00 |

ภาพที่ 4.3

กราฟเปรียบเทียบผลการรู้จำตามค่า N-best จำนวน 10 ประโยค



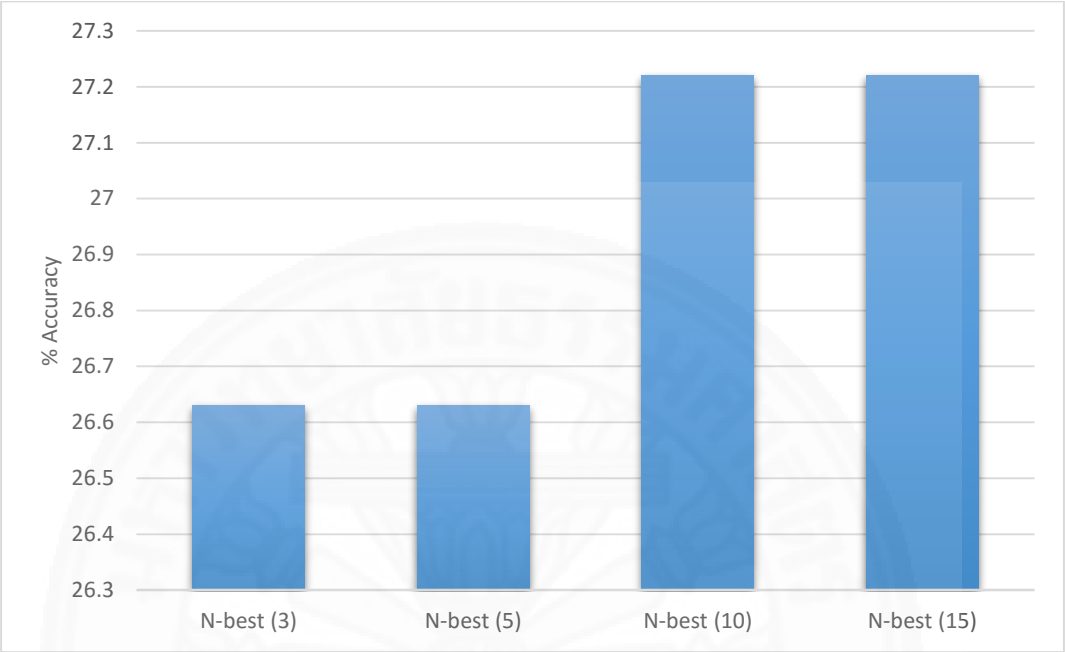

ตารางที่ 4.4

ภาพรวมเปรียบเทียบผลการรู้จำตามค่า N-best

| ค่า N-best | ความถูกต้อง |
|---|---|
| N-best (3) | 26.63 |
| N-best (5) | 26.63 |
| N-best (10) | 27.22 |
| N-best (15) | 27.22 |

ภาพที่ 4.4

กราฟภาพรวมเปรียบเทียบผลการรู้จำตามค่า N-best




จากการทดลองเพิ่มเติมเกี่ยวกับการปรับค่า parameter N-best เพื่อดูผลว่าจะได้ค่าความถูกต้องที่ดีขึ้นหรือไม่ โดยได้ทำการทดลองปรับค่า N-best เป็น 3, 5, 10 และ 15 ตามลำดับ โดยใช้ประโยค 10 ประโยคในการทดลองในแต่ละค่า N-best ซึ่งได้ผลการทดลองดังตารางที่ 4.3 และ ภาพที่ 4.3 ซึ่งผลลัพธ์ที่ได้นั้น ค่าความถูกต้องของ N-best 3, 5, 10 และ 15 นั้น ให้ค่าความถูกต้องที่เท่ากันในทุกประโยค ยกเว้นประโยคที่ 1 ที่ค่า N-best 10 และ 15 นั้น จะให้ค่าความถูกต้องที่มากกว่า ซึ่งในภาพรวมของผลลัพธ์ที่ได้นั้น ค่า N-best 3 และ 5 จะได้ค่าความถูกต้องที่ 26.63% ส่วนค่า N-best 10 และ 15 จะได้ค่าความถูกต้องที่ 27.22% ดังแสดงในตารางที่ 4.4 และภาพที่ 4.4 จากผลการทดลองพบว่าค่า N-best ที่ 10 ขึ้นไปนั้น จะช่วยให้ค่าความถูกต้องที่ดีขึ้นแต่ผลที่ได้นั้นดีขึ้นเพียงเล็กน้อย

จากการทดลองการปรับค่า parameter N-best นี้ ได้ทำการสรุปข้อมูลเพิ่มเติมในเรื่องเวลาที่ใช้ในการประมวลผลด้วยอัลกอริทึม N-best LIMABEAM เอาไว้ โดยทำการเปรียบเทียบเวลาที่ใช้ตามค่า parameter N-best ดังสรุปในตารางที่ 4.5 และ ภาพที่ 4.5 ซึ่งจากเวลาที่ได้จะเห็นว่า N-best 3 จะใช้เวลาในการประมวลผลอยู่ที่ 2 ชั่วโมง 16 นาที N-best 5 อยู่ที่ 6 ชั่วโมง 37 นาที เวลาเพิ่มขึ้นเป็นสัดส่วน 2.95 ส่วน N-best 10 ใช้เวลา 7 ชั่วโมง 26 นาที เวลา

เพิ่มขึ้นจาก N-best 5 คิดเป็นสัดส่วน 1.14 เท่า และ N-best 15 ใช้เวลา 8 ชั่วโมง 23 นาที เวลาเพิ่มขึ้นจาก N-best 10 คิดเป็นสัดส่วน 1.13 เท่า ซึ่งจะเห็นว่าเวลาที่ใช้ เพิ่มขึ้นจาก N-best 3 ไป N-best 5 มีเปอร์เซ็นต์ที่สูงกว่า เวลาที่เพิ่มขึ้นจาก N-best 5 ไป N-best 10 และ N-best 15

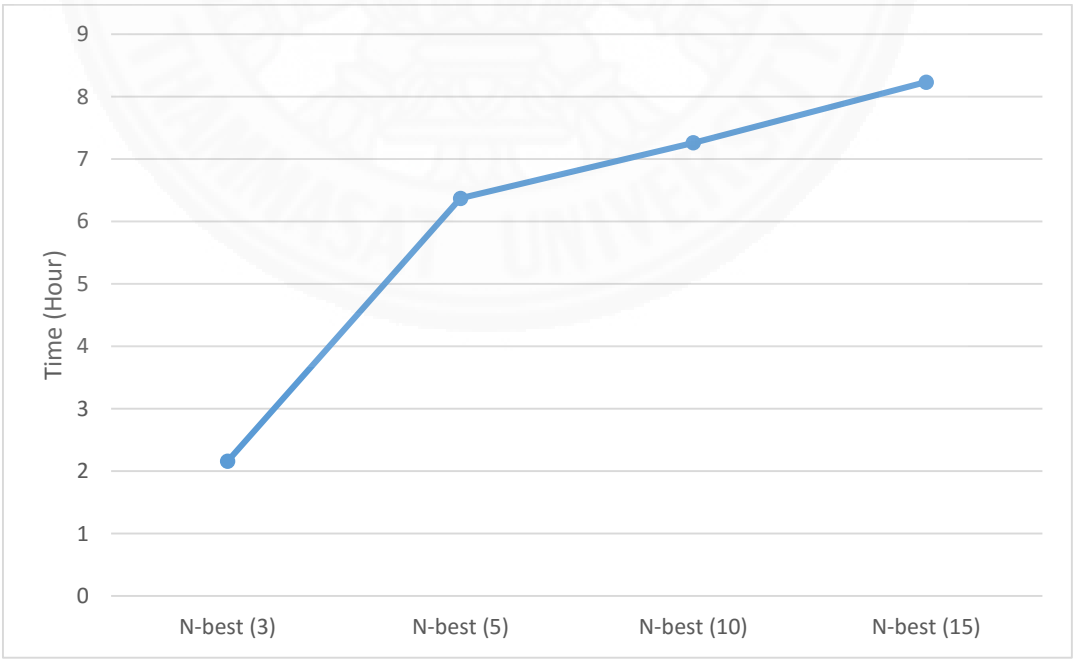
ตารางที่ 4.5

เปรียบเทียบเวลาที่ใช้ในการประมวลผลตามค่า N-best

| ค่า N-best | เวลา (ชั่วโมง) | อัตราส่วน |
|---|---|---|
| N-best (3) | 2.16 | - |
| N-best (5) | 6.37 | 2.95 |
| N-best (10) | 7.26 | 1.14 |
| N-best (15) | 8.23 | 1.13 |

ภาพที่ 4.5

กราฟเปรียบเทียบเวลาที่ใช้ในการประมวลผลตามค่า N-best



ตารางที่ 4.6

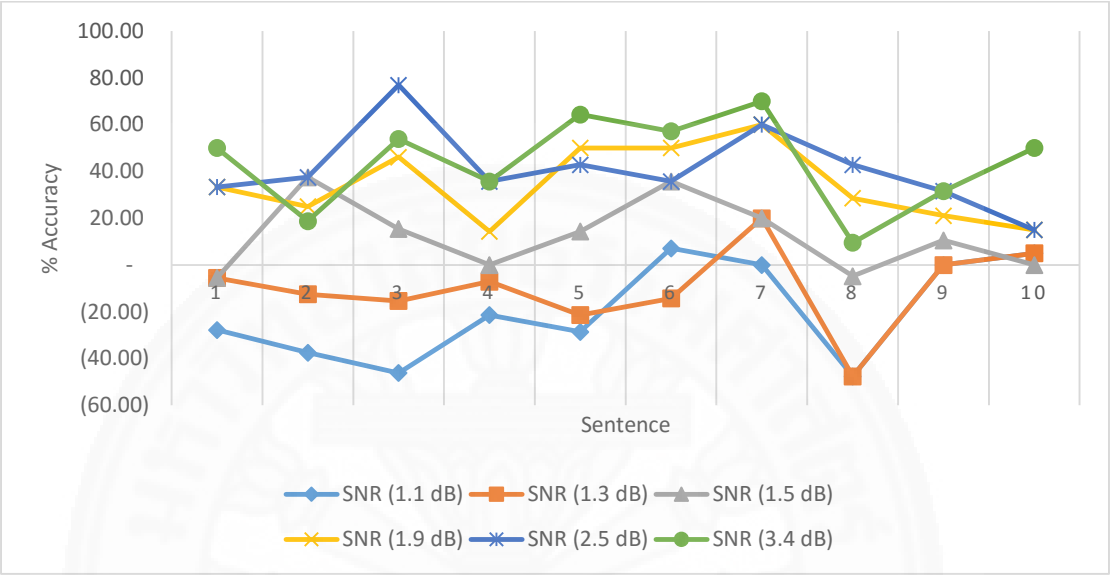
เปรียบเทียบผลการรู้จำตามค่าคุณภาพสัญญาณเสียงต่อเสียงรบกวน (SNR) จำนวน 10 ประโยค

| ประโยค | SNR (1.1 dB) | SNR (1.3 dB) | SNR (1.5 dB) | SNR (1.9 dB) | SNR (2.5 dB) | SNR (3.4 dB) |
|---|---|---|---|---|---|---|
| 1 | -27.78 | -5.56 | -5.56 | 33.33 | 33.33 | 50.00 |
| 2 | -37.50 | -12.50 | 37.50 | 25.00 | 37.50 | 18.75 |
| 3 | -46.15 | -15.38 | 15.38 | 46.15 | 76.92 | 53.85 |
| 4 | -21.43 | -7.14 | 0.00 | 14.29 | 35.71 | 35.71 |
| 5 | -28.57 | -21.43 | 14.29 | 50.00 | 42.86 | 64.29 |
| 6 | 7.14 | -14.29 | 35.71 | 50.00 | 35.71 | 57.14 |
| 7 | 0.00 | 20.00 | 20.00 | 60.00 | 60.00 | 70.00 |
| 8 | -47.62 | -47.62 | -4.76 | 28.57 | 42.86 | 9.52 |
| 9 | 0.00 | 0.00 | 10.53 | 21.05 | 31.58 | 31.58 |
| 10 | 5.00 | 5.00 | 0.00 | 15.00 | 15.00 | 50.00 |

จากการทดลองเปรียบเทียบผลการรู้จำด้วยอัลกอริทึม N-best LIMABEAM ตามค่าคุณภาพสัญญาณเสียงต่อเสียงรบกวน (SNR) ใช้ค่า N-best 3 ในการประมวลผล และใช้จำนวนประโยค 10 ประโยคในการทดลอง แยกตามค่า SNR จำนวน 6 ค่าด้วยกัน ซึ่งค่า SNR ต่ำแสดงว่ามีสัญญาณเสียงรบกวน (Noise) มาก ส่วนค่า SNR สูงแสดงว่ามีสัญญาณรบกวนน้อย ซึ่งผลที่ได้จากการทดลองแสดงในตารางที่ 4.6 และ ภาพที่ 4.6 ผลที่ได้จะเห็นว่าค่าความถูกต้องในบางประโยคของแต่ละค่า SNR ที่สูงกว่า บางทีก็ให้ค่าความถูกต้องที่ต่ำกว่าได้ ซึ่งน่าจะเป็นผลมาจากการใช้ชุดเสียงที่แตกต่างกันในแต่ละค่า SNR ซึ่งเกิดจากการพูดที่แตกต่างกันไปในแต่ละครั้ง

ภาพที่ 4.6

กราฟเปรียบเทียบผลการรู้จำตามค่าคุณภาพสัญญาณเสียงต่อเสียงรบกวน (SNR) จำนวน 10 ประโยค



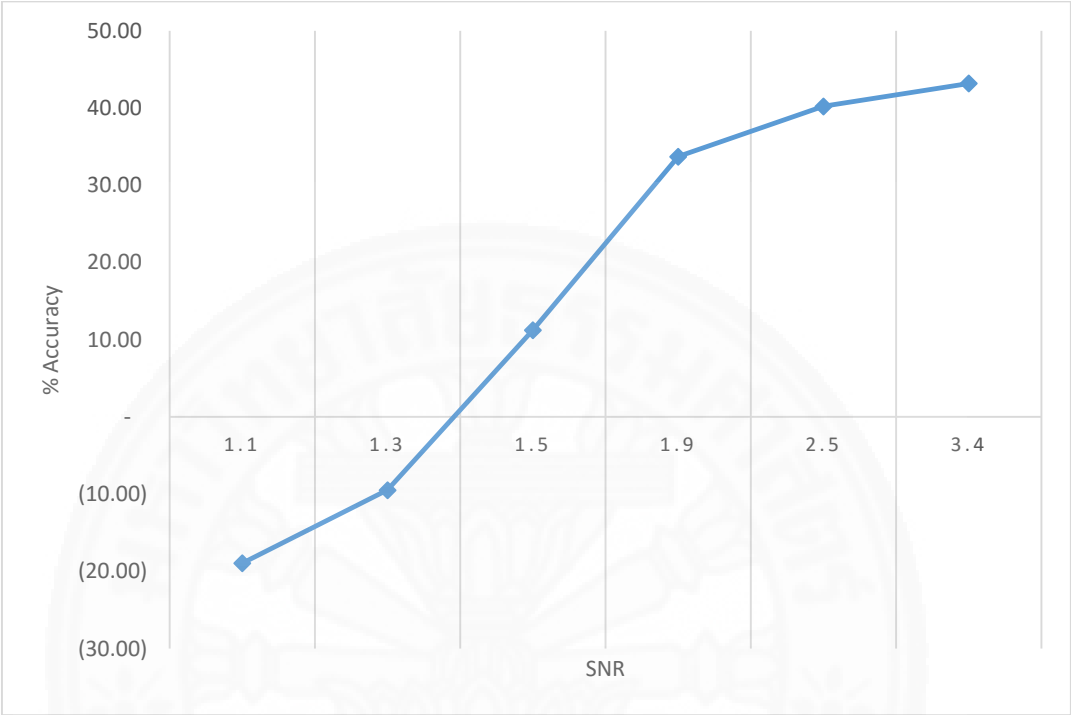

ตารางที่ 4.7

ภาพรวมเปรียบเทียบผลการรู้จำตามค่าคุณภาพสัญญาณเสียงต่อเสียงรบกวน (SNR)

| ค่าคุณภาพสัญญาณเสียงต่อเสียงรบกวน (SNR) | ค่าความถูกต้อง |
|---|---|
| 1.1 dB | -18.93 |
| 1.3 dB | -9.97 |
| 1.5 dB | 11.24 |
| 1.9 dB | 33.73 |
| 2.5 dB | 40.24 |
| 3.4 dB | 43.20 |

ภาพที่ 4.7

กราฟภาพรวมเปรียบเทียบผลการรู้จำตามค่าคุณภาพสัญญาณเสียงต่อเสียงรบกวน (SNR)




จากตารางที่ 4.7 และ ภาพที่ 4.7 เป็นการสรุปภาพรวมการเปรียบเทียบผลการรู้จำตามค่าคุณภาพสัญญาณเสียงต่อเสียงรบกวน (SNR) ซึ่งผลโดยค่ารวมที่ได้จากการทดลองทั้ง 10 ประโยคในแต่ละค่า SNR นั้น SNR 3.4 dB ให้ค่าความถูกต้องที่ดีที่สุดที่ 43.20% ส่วนค่า SNR ที่รองลงมาก็จะให้ค่าความถูกต้องที่ต่ำลงไปตามลำดับ จนถึงค่า SNR 1.1 dB ที่ให้ค่าความถูกต้องต่ำที่สุดที่ -18.93% ซึ่งก็แสดงให้เห็นว่ายิ่งมีสัญญาณรบกวนมากขึ้นเท่าไร ค่าความถูกต้องก็ยิ่งลดต่ำลงเรื่อย ๆ

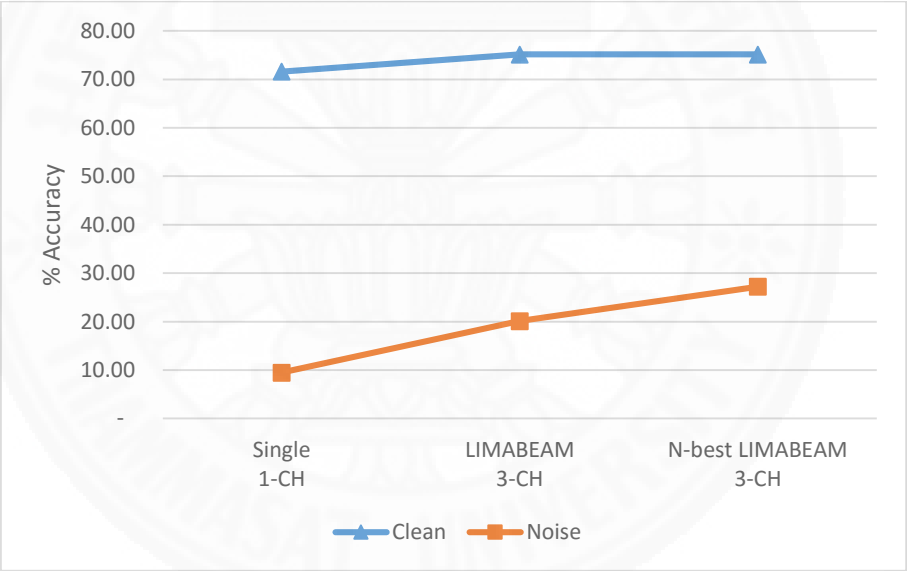
ตารางที่ 4.8

เปรียบเทียบการทดลองตามประเภทช่องรับสัญญาณ

| | Single 1-CH | LIMABEAM 3-CH | N-best LIMABEAM 3-CH |
|---|---|---|---|
| Clean | 71.60 | 75.15 | 75.15 |
| Noise | 9.47 | 20.12 | 27.22 |

ภาพที่ 4.8

กราฟเปรียบเทียบการทดลองตามประเภทช่องรับสัญญาณ




จากตารางที่ 4.8 และ ภาพที่ 4.8 เป็นการเปรียบเทียบการทดลองตามประเภทช่องรับสัญญาณระหว่างการใช้ช่องรับสัญญาณช่องทางเดียว (Single Channel) และ ช่องรับสัญญาณหลายช่องทาง (Multiple Channel) โดยใช้อัลกอริทึม LIMABEAM และ N-best LIMABEAM เป็นตัวเปรียบเทียบ และได้ทดลองเปรียบเทียบในชุดสัญญาณเสียง 2 ลักษณะด้วยกัน คือ ชุดสัญญาณเสียงในสภาพแวดล้อมที่ไม่มีเสียงรบกวน (Clean) และชุดสัญญาณเสียงในสภาพแวดล้อมที่มีเสียงรบกวน (Noise) โดยจากผลการทดลองที่ได้จะเห็นว่าค่าความถูกต้องในการใช้ช่องรับสัญญาณช่องทางเดียว จะให้ค่าความถูกต้องที่ต่ำกว่าการใช้ช่องรับสัญญาณแบบ

หลายช่องทาง ไม่ว่าจะใช้ชุดสัญญาณเสียงที่ไม่มีเสียงรบกวนหรือมีเสียงรบกวนก็ตาม แต่ค่าความถูกต้องของการใช้ชุดสัญญาณเสียงแบบไม่มีเสียงรบกวน ในแบบช่องทางเดียวนั้น จะให้ผลที่ไม่แตกต่างกันมากนักกับแบบหลายช่องทางทั้ง 2 อัลกอริทึม ที่ 71.60% และ 75.15% ในส่วนนี้อัลกอริทึม LIMABEAM และ N-best LIMABEAM ให้ค่าที่เท่ากัน ส่วนการใช้ชุดสัญญาณเสียบแบบมีเสียงรบกวน ในแบบช่องทางเดียว จะให้ผลที่แตกต่างกว่ามากกับการใช้แบบหลายช่องทาง โดยที่แบบช่องทางเดียวให้ค่าที่ 9.47% ส่วนแบบหลายช่องทางอัลกอริทึม LIMABEAM ได้ 20.12% และอัลกอริทึม N-best LIMABEAM ได้ค่าสูงสุดในกลุ่มนี้ที่ 27.22%

# บทที่ 5

# สรุปผลการวิจัยและข้อเสนอแนะ

## 5.1 การอภิปราย

งานวิจัยนี้เป็นการทดลองกับเสียงพูดภาษาไทยที่เป็นประโยคบทสนทนาในสภาพแวดล้อมที่มีเสียงรบกวน ซึ่งที่ผ่านมายังไม่มีการทำวิจัยในมุมมองนี้มาก่อน จากผลการทดลองอัลกอริทึม N-best LIMABEAM ให้ค่าความถูกต้องแม่นยำดีที่สุดที่ 27.22% ซึ่งพัฒนาได้ดีขึ้นจากอัลกอริทึม LIMABEAM แต่ค่าความถูกต้องแม่นยำที่ได้ยังค่อนข้างต่ำอยู่ น่าจะเป็นผลมาจากฐานข้อมูลที่ใช้ในการฝึกฝนเพื่อรู้จำเสียงมีจำนวนที่น้อยเลยทำให้ได้โมเดลเสียงที่ไม่ละเอียดเพียงพอที่จะจำแนก Observation sequence ที่ส่งเข้าไปจำแนกเสียงว่าเป็นเสียงของ phone ที่ถูกต้องที่สุด แต่อย่างไรก็ตามจากการทดลองในงานวิจัยนี้ ก็ยังให้ค่าความถูกต้องที่ดีกว่าอันเนื่องจากการใช้อัลกอริทึม N-best LIMABEAM ที่เป็นการเพิ่มโอกาสที่จะได้ค่าสมมุติฐานของ transcription ที่ดีที่สุดจากจำนวนชุดค่าสมมุติฐานที่ได้จากการทำ N-best ซึ่งวิธีนี้ก็จะช่วยเพิ่มผลลัพธ์ที่ได้ ให้ได้ค่าความถูกต้องมากยิ่งขึ้น จากการทดลองยังพบว่าหากยิ่งเพิ่มระดับความดังของเสียงรบกวน คือค่า SNR ที่มีค่าต่ำมาก ก็จะยิ่งทำให้ค่าความถูกต้องลดต่ำลงเรื่อย ๆ การปรับค่า parameter N-best ก็มีส่วนที่ทำให้ได้ผลลัพธ์ที่แตกต่างกัน จากผลการทดลองพบว่าค่า N-best ที่ 10 ขึ้นไปนั้น จะช่วยให้ค่าความถูกต้องที่ดีขึ้น แต่ผลที่ได้นั้นดีขึ้นเพียงเล็กน้อย อีกทั้งเวลาที่ใช้ในการประมวลผลที่มีสัดส่วนการเพิ่มขึ้นถึง 2.95 เท่า จากค่า N-best 3 ไป N-best 5 ส่วนสัดส่วนการเพิ่มขึ้นจาก N-best 5 ไป N-best 10 และ 15 มีค่าที่ใกล้เคียงกันอยู่ที่ 1.14 และ 1.13 เท่าตามลำดับ และยังพบว่าช่องทางการรับสัญญาณเสียงแบบหลายช่องทางก็จะให้ค่าความถูกต้องที่มากกว่าการรับสัญญาณเสียงช่องทางเดียว ทั้งในชุดข้อมูลที่มีเสียงรบกวน และไม่มีเสียงรบกวน

จากการทดลองนี้ยังมีข้อจำกัดอยู่ โดยมีปัญหาจากการบันทึกเสียงในขั้นตอนการฝึกฝน และ ขั้นตอนการรู้จำ หากมีการออกเสียงที่ไม่ชัดเจน ก็จะทำให้ได้ผลลัพธ์ที่ไม่ดี และแตกต่างกันออกไป การใช้ Smartphone เป็นไมโครโฟนในการบันทึกเสียง พบว่า Smartphone แต่ละเครื่องให้คุณภาพเสียงที่แตกต่างกัน อาจทำให้ได้คุณภาพเสียงหลังจากการรวมเสียงที่แตกต่างกัน การใช้เสียงในการทดสอบการรู้จำคนละเสียงกับที่ใช้ในการฝึกฝนก็มีผลทำให้ได้ค่าความ

ถูกต้องที่ไม่ดี ซึ่งน่าจะเป็นผลมาจากฐานข้อมูลที่ใช้ในการฝึกฝนเพื่อรู้จำเสียงมีจำนวนที่น้อยและมีลักษณะของเสียงไม่หลากหลาย

## 5.2 ข้อเสนอแนะในการทำวิจัยครั้งต่อไป

จากการทดลองนั้น พบว่ามีสิ่งที่ควรจะทดลองทำ หรือปรับปรุงให้ดีขึ้นได้ในงานวิจัยในอนาคต โดยสามารถสรุปได้ดังนี้

1. สามารถที่จะเพิ่มจำนวนไมโครโฟนให้มีมากยิ่งขึ้น ซึ่งในงานวิทยานิพนธ์ฉบับนี้ใช้จำนวนไมโครโฟนเพียง 3 ชุด

2. เปลี่ยนเครื่องบันทึกเสียงจาก Smartphone เป็นชุดไมโครโฟนชนิดเดียวกันเพื่อแก้ปัญหาการได้มาซึ่งคุณภาพเสียงที่แตกต่างกัน

3. พัฒนากระบวนการในการรู้จำให้ใช้เวลาในการประมวลผลเร็วขึ้น เนื่องจากในการทดลองนี้ จะใช้เวลาในการประมวลผลที่นาน ซึ่งถ้าสามารถทำเวลาได้เร็วขึ้นก็จะยิ่งดี

4. ทำการทดสอบกับสภาพแวดล้อมที่หลากหลายมากขึ้น เพื่อให้สามารถรู้จำเสียงได้ในทุกสภาพแวดล้อม

5. เพิ่มขนาดฐานข้อมูลเสียงที่ใช้ฝึกฝนให้มากขึ้น เพื่อให้สามารถรองรับกับเสียงได้หลายบุคคลมากขึ้น รวมทั้งช่วยให้ได้ค่าความถูกต้องแม่นยำที่ดีขึ้น

6. ให้สามารถรู้จำเสียงพูดได้หลากหลายบุคคล คือเป็นการรู้จำเสียงพูดที่ไม่ขึ้นกับเสียงผู้พูดเอง

# รายการอ้างอิง

## สื่ออิเล็กทรอนิกส์

ชัย วุฒิวิวัฒน์ชัย, สุทัศน์ แซ่ตั้ง, และ วารินทร์ อัจฉริยะกุลพร. (2543). ความก้าวหน้าของการพัฒนาระบบระบุผู้พูดภาษาไทย. NECTEC Technical Journal, 7, 24-35. สืบค้นเมื่อวันที่ 15 เมษายน 2556, จาก https://www.nectec.or.th/NTJ/No7/papers/No7_full_2.pdf

ณัฐนันท์ ทัดพิทักษ์กุล, และ บุญธีร์ เครือตราชู. (ม.ป.ป.). การลดสัญญาณรบกวนของการจดจำเสียงพูดด้วยการแปลงเวฟเล็ตโดยการประมาณค่าจุดเปลี่ยน. ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ. สืบค้นเมื่อวันที่ 14 ตุลาคม 2556, จาก http://tvis.nectec.or.th/speech/download/software/view/eecom27_ds004.pdf

บุญเสริม กิจศิริกุล, และ ณัฐกร ทับทอง. (2548). การพัฒนาระบบการรู้จำเสียงพูดภาษาไทย. สืบค้นเมื่อวันที่ 12 ตุลาคม 2556, จาก http://www.cp.eng.chula.ac.th/~boonserm/publication/ThaiASR_kt2005.pdf

ภัชริกา คชสำโรง, ตรีภพ สรรเพชรนิยม, ศวิต กาสุริยะ, ณัฐนันท์ ทัดพิทักษ์กุล, และ ชัย วุฒิวิวัฒน์ชัย. (ม.ป.ป.). ฐานข้อมูลเสียงขนาดใหญ่สำหรับระบบรู้จำเสียงพูดต่อเนื่องภาษาไทย. ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ. สืบค้นเมื่อวันที่ 23 ตุลาคม 2556, จาก http://tvis.nectec.or.th/speech/download/software/view/nac2005.pdf

มนตรี โพธิโสโนทัย, และ เฉลิมภัณฑ์ ฟองสมุทร. (2554). วิธีการรู้จำเสียงพูดภาษาไทยแบบทนทานต่อเสียงรบกวนภายนอก. วารสารเทคโนโลยีสารสนเทศ, 7(13), 19-23. สืบค้นเมื่อวันที่ 12 ตุลาคม 2556, จาก http://www.it.kmutnb.ac.th/journal/pdf/vol13/ch04.pdf

Aarts, M., Pries, H., & Doff, A. (2012). Two Sensor Array Beamforming Algorithm for Android Smartphones. Retrieved October 12, 2013, from http://repository.tudelft.nl/assets/uuid:7b7b6fda-3446-49ee-84b0-4b7540914b80/Two_Sensor_Beamforming_Algorithm.pdf

Acero, A. (1990). Acoustical and Environmental Robustness in Automatic Speech Recognition. Retrieved April 15, 2013, from http://www.msr-waypoint.com/pubs/78776/acero_thesis.pdf

Acero, A., & Stern, R. M. (n.d.). Robust Speech Recognition by Normalization of the Acoustic Space. Retrieved April 15, 2013, from http://www.cs.cmu.edu/~robust/Papers/ica91.pdf

Brayda, L., Wellekens, C., & Omologo, M. (2006). Improving robustness of a likelihood-based beamformer in a real environment for automatic speech recognition. Retrieved October 12, 2013, from http://www.eurecom.fr/en/publication/2054/download/mm-braylu-060626.pdf

Kim, C., & Stern, R. M. (2010). Nonlinear Enhancement of Onset for Robust Speech Recognition. INTERSPEECH 2010, 2058-2061. Retrieved April 15, 2013, from https://pdfs.semanticscholar.org/c1a2/88c0d92f35ffe6ec7a11d53da877971fbe16.pdf

Kleinschmidt, T. F. (2010). Robust Speech Recognition using Speech Enhancement. Retrieved February 25, 2014, from http://eprints.qut.edu.au/31895/1/Tristan_Kleinschmidt_Thesis.pdf

Pisarn, C., & Theeramunkong, T. (2007). Thai spelling analysis for automatic spelling speech recognition. Information Sciences, 2008(178), 122-136. Retrieved October 12, 2013, from https://pdfs.semanticscholar.org/a774/a3eee08d0b64b98c761e1dc9d9ab1eeaac80.pdf

Rabiner, L., & Juang, B.-H. (1993). Fundamentals of speech recognition. New Jersey: Prentice Hall. Retrieved October 12, 2013, from http://www.cin.ufpe.br/~ags/An%E1lise%20de%20voz%20e%20v%EDdeo/Fundamentals_of_Speech_Recognition.pdf

Raj, B., Seltzer, M. L., & Reyes, M. (2003). Speech Recognizer based Maximum Likelihood Beamforming. Retrieved February 24, 2014, from

https://pdfs.semanticscholar.org/1305/91988467bb4b4fd95ea713c142e2bf5a35f9.pdf

Raj, B., Singh, R., & Stern, R. M. (2004). On Tracking Noise with Linear Dynamical System Models. ICASSP 2004. Retrieved April 15, 2013, from http://www.cs.cmu.edu/~robust/Papers/icassp04raj.pdf

Rotili, R., Piazza, F., & Chiaraluce, F. (2012). Multi-channel Algorithms for Improving Speech Recognition Accuracy in Adverse Environments. Retrieved January 18, 2014, from http://openarchive.univpm.it/jspui/bitstream/123456789/475/1/Tesi.Rotili.pdf

Seltzer, M. L. (2003). Microphone Array Processing for Robust Speech Recognition. Retrieved October 12, 2013, from http://www.cs.cmu.edu/~robust/Thesis/mseltzer_phdthesis.pdf

Seltzer, M. L., Raj, B., & Stern, R. M. (2004). Likelihood-Maximizing Beamforming for Robust Hands-Free Speech Recognition. Retrieved February 24, 2014, from http://www.merl.com/publications/docs/TR2004-088.pdf

Shan, J., Wu, G., Hu, Z., Tang, X., Jansche, M., & Moreno, P. J. (2010). Search by Voice in Mandarin Chinese. INTERSPEECH 2010, 354-357. Retrieved October 12, 2013, from http://static.googleusercontent.com/media/research.google.com/th//pubs/archive/36463.pdf

Tarsaku, P., & Kanokphara, S. (n.d.). A study of HMM-based automatic segmentations for Thai continuous speech recognition system. National Electronics and Computer Technology Center (NECTEC). Retrieved December 23, 2013, from http://tvis.nectec.or.th/speech/download/software/view/snlp2002.pdf

Thatphithakkul, N., & Kanokphara, S. (2004). HMM Parameter Optimization Using TABU Search. ISCIT 2004, 904-908. Retrieved December 23, 2013, from http://spt.nectec.or.th/speech/download/software/view/29am1c-3.pdf

**สื่ออิเล็กทรอนิกส์อื่น ๆ**

ณรงค์รัตน์ เลี้ยวรุ่งโรจน์, อนุพงษ์ ธรรมรักษาสิทธิ์, และ โกสินทร์ จำนงไทย. (2546). รถเข็นคนพิการควบคุมด้วยระบบรู้จำเสียงพูด [เว็บไซต์]. สืบค้นเมื่อวันที่ 15 เมษายน 2556, จาก http://www.kmutt.ac.th/rippc/best47.htm

หน่วยปฏิบัติการวิจัยวิทยาการมนุษยภาษา. (ม.ป.ป.). Thai speech recognition tutorial [เว็บไซต์]. ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ. สืบค้นเมื่อวันที่ 12 ตุลาคม 2556, จาก http://tvis.nectec.or.th/speech/index.php?Itemid=78

# ประวัติผู้เขียน

| | |
|---|---|
| ชื่อ | นายรุ่งโรจน์ กอปัญญาพิพัฒน์ |
| วันเดือนปีเกิด | 9 ธันวาคม 2518 |
| วุฒิการศึกษา | ปริญญาตรี<br>สาขาวิทยาการคอมพิวเตอร์<br>คณะวิทยาศาสตร์<br>มหาวิทยาลัยมหิดล |
| ประสบการณ์ทำงาน | 2554 - ปัจจุบัน, Chanwanich Co., Ltd.,<br>Project Delivery Manager<br>2550 - 2554, Progress Software Co., Ltd.,<br>Project Manager<br>2547 - 2550, Cubic Solutions Co., Ltd.,<br>Project Manager<br>2543 - 2547, Mweb (Thailand) Co., Ltd.,<br>Senior Consultant Developer<br>2542 - 2543, Internet Info Network Co., Ltd.,<br>Senior Developer<br>2541 - 2542, Gemkey (Thailand) Co., Ltd.,<br>Programmer |