การค้นคืนข้ามภาษาสำหรับคำทับศัพท์ภาษาไทย/อังกฤษด้วยวิธีการนิวรอลเน็ตเวิร์กแบบจำลองฮิดเด็นมาร์คอฟ และขั้นตอนวิธีเชิงพันธุกรรม

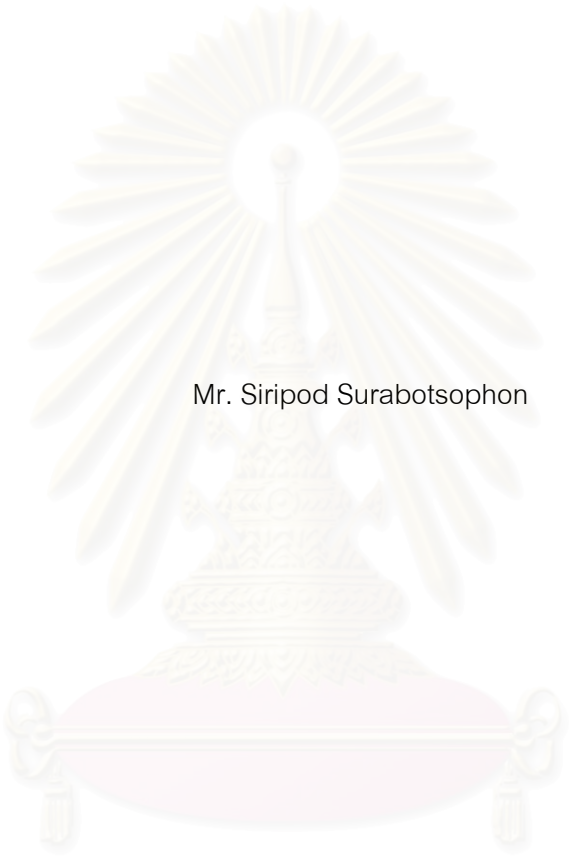
นาย ศิริพจน์ สุรบถโสภณ

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2547

ISBN 974-17-6383-2

THAI/ENGLISH CROSS-LANGUAGE TRANSLITERATED WORD RETRIEVAL USING NEURAL NETWORKS, HIDDEN MARKOV MODELS, AND GENETIC ALGORITHMS

Mr. Siripod Surabotsophon



A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering in Computer Engineering

Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University

Academic Year 2547

ISBN 974-17-6383-2

หัวข้อวิทยานิพนธ์	การค้นคืนข้ามภาษาสำหรับคำทับศัพท์ภาษาไทย/อังกฤษด้วยวิธีการนิวรอลเน็ตเวิร์ก แบบจำลองฮิดเด็นมาร์คอฟ และขั้นตอนวิธีเชิงพันธุกรรม
โดย	นาย ศิริพจน์ สุรบถโสภณ
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.ดิเรก ลาวัณย์ศิริ)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(รองศาสตราจารย์ ดร.ประภาส จงสถิตย์วัฒนา)

..... อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล)

.....กรรมการ
(รองศาสตราจารย์ ดร.สมชาย ประสิทธิ์จูตระกูล)

.....กรรมการ
(รองศาสตราจารย์ ดร.ธนารักษ์ ธีระมั่นคง)

นาย ศิริพจน์ สุรบถโสภณ : การค้นคืนข้ามภาษาสำหรับคำทับศัพท์ภาษาไทย/อังกฤษด้วยวิธีการนิวรอลเน็ตเวิร์ก แบบจำลองฮิดเด็นมาร์คอฟ และขั้นตอนวิธีเชิงพันธุกรรม. (THAI/ENGLISH CROSS-LANGUAGE TRANSLITERATED WORD RETRIEVAL USING NEURAL NETWORKS, HIDDEN MARKOV MODELS, AND GENETIC ALGORITHMS) อ. ที่ปรึกษา : ผศ.ดร.บุญเสริม กิจศิริกุล, 61 หน้า. ISBN 974-17-6383-2.

วิทยานิพนธ์ฉบับนี้นำเสนอการการค้นคืนข้ามภาษาสำหรับคำทับศัพท์ภาษาไทย/อังกฤษโดยใช้วิธีการนิวรอลเน็ตเวิร์กและแบบจำลองฮิดเด็นมาร์คอฟในการเข้ารหัสคำ และใช้ขั้นตอนวิธีเชิงพันธุกรรม เพื่อเพิ่มความถูกต้องของการค้นคืน วิธีการที่นำเสนอช่วยให้สามารถค้นคืนคำทับศัพท์ข้ามภาษาได้โดยไม่ต้องอาศัยพจนานุกรม

ในการค้นคืนข้ามภาษาโดยไม่อาศัยพจนานุกรมนั้นจำเป็นต้องใช้หลักการเข้ารหัสซึ่งเป็นสัญลักษณ์แทนเสียงอ่านของคำและประกอบด้วยรหัสเสียงของแต่ละตัวอักษรของคำมาเรียงต่อกัน ในการที่จะทราบว่าตัวอักษรที่กำลังสนใจในคำนั้นให้รหัสเสียงใดจำเป็นต้องอาศัยการพิจารณาตัวอักษรข้างเคียงด้วย ดังนั้นการเข้ารหัสคำสามารถจัดได้ว่าเป็นปัญหาการจำแนกอย่างหนึ่ง ด้วยเหตุนี้จึงได้นำวิธีการนิวรอลเน็ตเวิร์กและแบบจำลองฮิดเด็นมาร์คอฟมาใช้ในการเข้ารหัสคำ แต่เนื่องจากว่ารหัสคำของคำไทยและอังกฤษที่มีเสียงอ่านตรงกัน อาจมีความแตกต่างกันบ้าง จึงได้ใช้ขั้นตอนวิธีเชิงพันธุกรรมเพื่อหาต้นทุนการแก้ไขอักขระที่ใช้ในเทคนิคการเปรียบเทียบแบบประมาณสำหรับการค้นคืนคำที่มีเสียงอ่านคล้ายกันมากที่สุด จากผลการทดลองด้วยวิธี K-fold cross validation พบว่าเมื่อใช้นิวรอลเน็ตเวิร์กร่วมกับขั้นตอนวิธีเชิงพันธุกรรมสามารถให้ผลการค้นคืน F1 ได้ประมาณ 90% และเมื่อใช้แบบจำลองฮิดเด็นมาร์คอฟกับขั้นตอนวิธีเชิงพันธุกรรมสามารถให้ผลการค้นคืน F1 ได้ประมาณ 80%

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต.....
สาขาวิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่ออาจารย์ที่ปรึกษา.....
ปีการศึกษา 2547 ลายมือชื่ออาจารย์ที่ปรึกษาร่วม.....

4570562421 : MAJOR COMPUTER ENGINEERING

KEY WORD: TRANSLITERATED WORD / INFORMATION RETRIEVAL / NEURAL NETWORKS / HIDDEN MARKOV MODELS / GENETIC ALGORITHM

SIRIPOD SURABOTSOPHON : THAI/ENGLISH CROSS-LANGUAGE TRANSLITERATED WORD RETRIEVAL USING NEURAL NETWORKS, HIDDEN MARKOV MODELS, AND GENETIC ALGORITHMS. THESIS ADVISOR : ASST. PROF. BOONSERM KIJSIRIKUL, Ph.D., 61 pp. ISBN 974-17-6383-2.

This thesis presents Thai/English cross-language transliterated word retrieval by using Neural Networks and Hidden Markov Models for encoding words and using the Genetic Algorithm for improving the efficiency of the retrieval. The proposed method enables the transliterated word retrieval without using the dictionary.

Without dictionary, the phonetic code is employed for cross-language retrieval. The phonetic code of a word represents the sound of the word and it consists of a sequence of phonetic codes of characters in the word. In order to determine the code of a particular character, it is necessary to consider its surrounding characters. Hence this problem can be identified as a classification problem. For this reason, Neural Networks and Hidden Markov Models are used in phonetic encoding. However, as the codes generated from a pair of corresponding Thai/English words are sometimes slightly different, the Genetic Algorithm is applied to determine the appropriate cost of character editing used in approximate string matching. The experimental results, using K-fold cross validation, show that the F1-measure of 90% can be obtained when using Neural Networks and the Genetic Algorithm, and of 80% when using Hidden Markov Models and the Genetic Algorithm.

Department Computer Engineering Student's.....

Field of study Computer Engineering Advisor's.....

Academic year 2547 Co-advisor's.....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยความช่วยเหลืออย่างดียิ่งของผู้ช่วยศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล โดยท่านได้ให้คำแนะนำและข้อคิดเห็นต่างๆในการวิจัยมาโดยตลอด รวมทั้งตรวจแก้วิทยานิพนธ์ฉบับนี้อย่างละเอียด รวมทั้งคณะกรรมการสอบวิทยานิพนธ์ทุกท่านที่ได้ให้ข้อเสนอแนะและแนวทางอันเป็นประโยชน์ยิ่งในการทำวิจัย

ขอขอบคุณสมาชิกห้องปฏิบัติการอัจฉริยภาพเครื่องจักรและการค้นพบความรู้ (MIND LAB) และบรรดาเพื่อนๆ รุ่นพี่ และรุ่นน้อง ที่ให้คำแนะนำในงานวิจัยและให้ความเอื้อเฟื้อเครื่องคอมพิวเตอร์เพื่อใช้ทำการทดลองในงานวิจัยนี้

ท้ายที่สุดนี้ผู้วิจัยขอกราบขอบพระคุณบิดา มารดา และทุกคนในครอบครัว ที่ให้การสนับสนุนและให้กำลังใจเสมอมา

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

หน้า

สารบัญตาราง (ต่อ)

หน้า

สารบัญภาพ

หน้า

สารบัญภาพ (ต่อ)

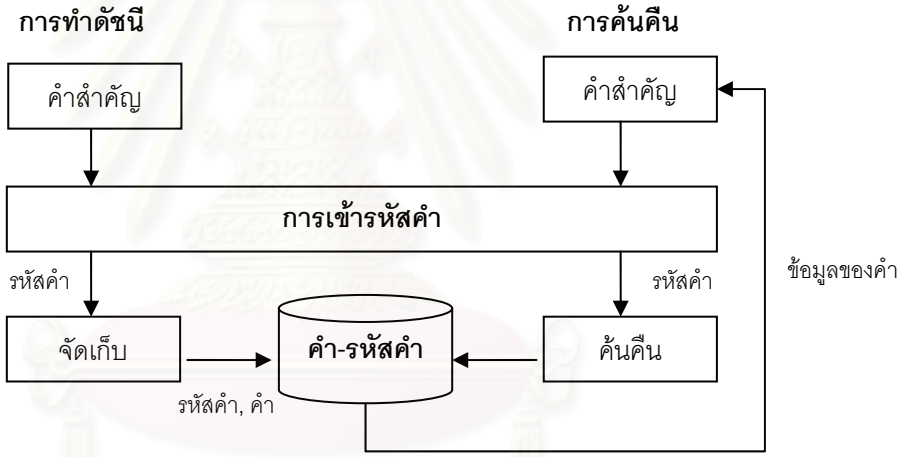
หน้า

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

การค้นคืนสารสนเทศข้ามภาษา (Cross–Language Information Retrieval) หมายถึง การค้นคืนสารสนเทศซึ่งภาษาที่แสดงในเอกสารไม่ตรงกับภาษาที่ใช้ในการสอบถาม ซึ่งเอกสารเหล่านี้โดยเฉพาะอย่างยิ่งเอกสารทางด้านวิทยาศาสตร์และวิศวกรรมโดยมากแล้วมักจะมีคำทับศัพท์ที่เป็นคำนามเฉพาะและคำศัพท์เทคนิคต่างๆเป็นจำนวนมาก ซึ่งคำทับศัพท์อาจจะเป็นคำอังกฤษทับศัพท์คำไทย หรือคำไทยทับศัพท์คำอังกฤษก็ได้ ดังนั้นระบบค้นคืนสารสนเทศข้ามภาษาจะมีประโยชน์อย่างมากในการค้นข้อมูลในลักษณะดังกล่าว



รูปที่ 1.1 การค้นคืนคำทับศัพท์ข้ามภาษาโดยอาศัยการเข้ารหัสคำ

ปัญหาในการค้นคืนคำทับศัพท์ข้ามภาษามีหลายประการโดยเฉพาะการที่คำในภาษาหนึ่งอาจจะถูกเขียนในอีกภาษาหนึ่งได้หลายรูปแบบ ตัวอย่างเช่น คำภาษาอังกฤษ “Interface” เมื่อเขียนทับศัพท์ในภาษาไทยอาจพบได้ทั้ง “อินเตอร์เฟส” หรือ “อินเตอร์เฟซ” หรือ คำภาษาไทย “เจริญ” อาจเขียนทับศัพท์ในภาษาอังกฤษเป็น “Charoen” หรือ “Jarern” ซึ่งวิธีการแก้ปัญหาวิธีหนึ่งคือการนำพจนานุกรมสองภาษามาใช้ในระบบค้นคืนสารสนเทศซึ่งก็ไม่สามารถแก้ปัญหาได้ตลอด เนื่องจากมีคำศัพท์เทคนิคใหม่มากมายในหลากหลายสาขาเกิดขึ้นอยู่เสมอ และคำเหล่านี้ส่วนมากมักไม่ปรากฏในพจนานุกรม ด้วยเหตุนี้จึงได้มีการใช้รหัสคำมาช่วยในการค้นคำทับศัพท์ข้ามภาษา โดยรหัสคำนี้จะเป็นสัญลักษณ์แทนเสียงอ่านของคำ คำที่มีเสียงอ่านตรงกันจะมีรหัสคำ

ที่ตรงกันหรือใกล้เคียงกัน รูปที่ 1.1 แสดงการค้นคืนคำทับศัพท์ข้ามภาษาโดยอาศัยการเข้ารหัสคำ ระบบจะทำการรวบรวมดัชนีคำสำคัญ (keyword) ต่างๆที่ปรากฏในเอกสารต่างๆไว้ แล้วนำคำสำคัญเหล่านี้ไปผ่านตัวเข้ารหัสคำเพื่อให้ได้รหัสคำออกมา จากนั้นนำไปจัดเก็บในฐานข้อมูล เมื่อมีผู้ใช้ต้องการค้นข้อมูล ก็ใส่คำสำคัญที่ต้องการโดยจะเป็นคนละภาษากับที่เก็บในเอกสารของระบบก็ได้ ระบบจะทำการแปลงคำนั้นให้เป็นรหัสคำแล้วนำไปค้นโดยการเทียบรหัสคำกับข้อมูลในฐานข้อมูล ถ้ารหัสคำตรงกันก็จะคืนข้อมูลคำนั้นกลับไปยังผู้ใช้

องค์ประกอบสำคัญสองส่วนในการค้นคืนคำทับศัพท์ข้าม ได้แก่ การเข้ารหัสคำและการเปรียบเทียบรหัสคำ ขั้นตอนวิธีการเข้ารหัสคำที่ดีจะช่วยให้รหัสคำที่ได้นั้นแทนเสียงอ่านของคำได้อย่างถูกต้องหรือใกล้เคียงความเป็นจริงมากที่สุด แต่ในบางครั้งนั้นอาจเป็นไปได้ว่ารหัสคำของคู่คำที่มีเสียงอ่านตรงกันจากทั้งสองภาษาอาจมีความแตกต่างกันบ้างเล็กน้อย การเปรียบเทียบรหัสคำจึงเป็นส่วนหนึ่งที่มีผลกระทบต่อการค้นคืนเนื่องจากการเปรียบเทียบรหัสคำเป็นเกณฑ์การตัดสินว่ารหัสคำที่นำมาเปรียบเทียบกันนั้นมีเสียงอ่านตรงกันหรือไม่ การเปรียบเทียบรหัสคำที่ไม่เหมาะสมสามารถทำให้การค้นคืนเกิดความผิดพลาดได้ เช่นบางครั้งอาจตัดสินว่าคำที่เปรียบเทียบกันนั้นมีเสียงอ่านไม่ตรงกัน ซึ่งทั้งที่จริงแล้วมีเสียงอ่านตรงกัน ดังนั้นในงานวิจัยนี้จึงมุ่งเน้นองค์ประกอบสองส่วนดังกล่าวในการค้นคืนคำทับศัพท์ข้ามภาษาไทย-อังกฤษ

1.2 วัตถุประสงค์ของการวิจัย

ออกแบบและพัฒนาวิธีการเข้ารหัสคำและการค้นคืนข้ามภาษาไทย-อังกฤษโดยใช้เทคนิคการเรียนรู้ของเครื่องแบบนิวรอลเน็ตเวิร์ก แบบจำลองฮิดเด็นมาร์คอฟ และขั้นตอนวิธีเชิงพันธุกรรม

1.3 ขอบเขตของการวิจัย

1. คำทับศัพท์ที่ใช้เป็นคำทับศัพท์ระหว่างภาษาไทยและภาษาอังกฤษเท่านั้น
2. คำศัพท์ในภาษาอังกฤษที่ใช้ไม่รวมถึงคำย่อ (Abbreviation) และคำรัสพจน์ (Acronym)
3. ยึดหลักเกณฑ์การออกเสียงของคำตามหลักของราชบัณฑิตยสถาน

1.4 ประโยชน์ที่คาดว่าจะได้รับ

สามารถนำไปใช้ในระบบการสืบค้นข้อมูลให้สามารถค้นคืนข้ามภาษาไทยและภาษาอังกฤษได้โดยไม่ต้องอาศัยพจนานุกรมและสามารถรองรับคำที่ศัพท์ที่เกิดขึ้นใหม่ได้ รวมทั้งสามารถใช้เป็นแนวทางในการสร้างระบบการค้นคืนข้ามภาษาในภาษาอื่นหรือการค้นคืนด้วยวิธีการที่ดียิ่งขึ้นได้

1.5 วิธีดำเนินการวิจัย

1. รวบรวมและจัดเก็บชุดข้อมูลคำทับศัพท์เพื่อใช้ในการทดลอง และกำหนดรหัสคำของแต่ละคำศัพท์ที่รวบรวมไว้
2. แปลงข้อมูลคำศัพท์ให้อยู่ในรูปที่ใช้สำหรับฝึกสอนนิวรอลเน็ตเวิร์ก และแบบจำลองฮิดเด็นมาร์คอฟ
3. ฝึกสอนและทดสอบนิวรอลเน็ตเวิร์ก และแบบจำลองฮิดเด็นมาร์คอฟ เพื่อใช้เป็นตัวสร้างรหัสคำ
4. ใช้ขั้นตอนวิธีเชิงพันธุกรรมเพื่อเพิ่มประสิทธิภาพของการค้นคืน
5. ทำการทดลองและปรับปรุงผลการทดลอง
6. สรุปผลการทดลอง และจัดทำวิทยานิพนธ์

1.6 ผลงานที่ตีพิมพ์จากงานวิจัย

ส่วนหนึ่งของวิทยานิพนธ์นี้ได้ตีพิมพ์และนำเสนอในงานประชุมวิชาการวิทยาการคอมพิวเตอร์และวิศวกรรมคอมพิวเตอร์แห่งชาติ 2546 (The National Computer Science and Engineering Conference: NCSEC'03) เมื่อวันที่ 28-30 ตุลาคม พ.ศ. 2546 ในบทความเรื่อง "การค้นคืนข้ามภาษาสำหรับคำทับศัพท์ภาษาไทย/อังกฤษด้วยวิธีการนิวรอลเน็ตเวิร์ก แบบจำลองฮิดเด็นมาร์คอฟ และขั้นตอนวิธีเชิงพันธุกรรม" (Thai-English Cross-Language Transliterated Word Retrieval Using Neural Networks, Hidden Markov Models, and Genetic Algorithms) โดยผู้นำเสนอคือ ศิริพจน์ สุรบถโสภณ และ บุญเสริม กิจศิริกุล

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 การถอดอักษร

การถอดอักษร (Transliteration) หมายถึง การนำคำในภาษาหนึ่งมาเขียนด้วยตัวอักษรอีกภาษาหนึ่งแบบอักษรต่ออักษร โดยพยายามใช้หน่วยเสียงของอักษรทั้งสองภาษาใกล้เคียงกันมากที่สุด [1] ตัวอย่างเช่น คำว่า “OXFORD” ในภาษาอังกฤษถอดอักษรเป็น “ออกซ์ฟอร์ด” ในภาษาไทย เป็นต้น การถอดอักษรแบ่งเป็น 3 ขั้นตอนหลัก ๆ ดังนี้

1. ถอดหน่วยอักษรในภาษาต้นแบบ (Source Language) เป็นหน่วยเสียงในภาษาต้นแบบ เช่น ถอดหน่วยอักษร “B” ในภาษาอังกฤษเป็นหน่วยเสียง /b/ ในภาษาอังกฤษ เป็นต้น
2. แทนหน่วยเสียงในภาษาต้นแบบ ด้วยหน่วยเสียงในภาษาเป้าหมาย (Target Language) โดยพยายามใช้หน่วยเสียงที่ใกล้เคียงกันมากที่สุด เช่น แทนหน่วยเสียง /b/ ในภาษาอังกฤษเป็นหน่วยเสียง /b/ ในภาษาไทย เป็นต้น
3. ถอดหน่วยเสียงในภาษาเป้าหมาย เป็นหน่วยอักษรในภาษาเป้าหมาย เช่น ถอดหน่วยเสียง /b/ ในภาษาไทยเป็นหน่วยอักษร “บ” ในภาษาไทย เป็นต้น

ปัญหาต่าง ๆ ในการถอดอักษรได้แก่

1. ความสัมพันธ์ของหน่วยอักษรและหน่วยเสียง มีความสัมพันธ์แบบหนึ่งตัวอักษรแทนหลายหน่วยเสียง เช่น ในภาษาอังกฤษ “C” แทนด้วย /k/ หรือ /s/ เป็นต้น และมีความสัมพันธ์แบบหลายหน่วยอักษรแทนหนึ่งหน่วยเสียง เช่น ในภาษาอังกฤษ “N, TN, GN, PN” แทนด้วย /n/ ในภาษาไทย “ร, ฤ, หร” แทนด้วย /r/ และ “ฉ, ช, ฌ” แทนด้วย /ch/ เป็นต้น
2. การแบ่งพยางค์ในภาษาต้นแบบ เมื่อมีพยัญชนะตัวเดียวอยู่ระหว่างสระ เช่น คำว่า money ในภาษาอังกฤษ จะแบ่งพยางค์อย่างไร จะถอดพยัญชนะซ้ำสองตัวเพื่อให้อ่านได้สะดวกเป็น มัน-นีย์ หรือจะถอดอักษรเพียงตัวเดียวตามที่ปรากฏในภาษาอังกฤษเป็น มะ-นีย์ หรือ มัน-อีย์
3. ปัญหาอันเนื่องมาจากช่วงเวลาของการยืมคำทับศัพท์ คำทับศัพท์บางคำยืมมาเป็นเวลานาน ซึ่งในอดีตมีหลักเกณฑ์การทับศัพท์ไม่ตรงกับหลักเกณฑ์ในปัจจุบัน เช่น “C” ที่แทน /k/ ในอดีตนิยมถอดเป็นอักษร “ก” เช่น กุ๊ก (Cook) กัปตัน (Captain) กะรัต(Carat)

แก๊ป(Cap) เป็นต้น แต่ปัจจุบัน “C” ที่แทน /k/ มักจะถอดเป็น “ค” ในตำแหน่งพยัญชนะต้น เช่น คอนโดมิเนียม (Condominium) แคปซูล (Capsule) แคร็อต (Carrot) เป็นต้น

2.2 การถ่ายเสียงด้วยตัวอักษรโรมัน

การถ่ายเสียงด้วยตัวอักษรโรมัน (Romanization) คือการถ่ายเสียงตัวอักษรของภาษาอื่นที่ไม่ใช่อักษรโรมัน เช่น ไทย จีน ญี่ปุ่น ฯลฯ ให้เป็นตัวอักษรโรมัน [1] เพื่อให้ผู้ที่ไม่รู้จักภาษานั้น ๆ สามารถอ่านออกเสียงได้ ทางราชบัณฑิตยสถานจึงได้กำหนดระบบการใช้ตัวอักษรโรมันเพื่อการถ่ายเสียงสำหรับตัวอักษรไทยออกเป็น 2 ระบบ คือระบบทั่วไปและระบบพิสดาร โดยระบบทั่วไปจะใช้สำหรับกรณีที่การออกเสียงสำคัญกว่าการเขียนตัวสะกด ซึ่งจะอาศัยหลักการออกเสียงเป็นสำคัญ ต้องสอดคล้องกับไวยากรณ์ของไทย และสามารถขยายเป็นระบบเฉพาะได้ เช่น คำว่า “กษัตริย์” ถ่ายเสียงเป็น Kasat ส่วนระบบพิสดารจะใช้ในกรณีที่จะแสดงตัวอักษรให้ละเอียดแม่นยำ เพื่อให้คงความหมายของคำนั้นไว้ เช่น คำว่า “กษัตริย์” ถ่ายเสียงเป็น Kasatriy

2.3 นิวรอลเน็ตเวิร์ก (Neural Networks)

นิวรอลเน็ตเวิร์กเป็นวิธีการเรียนรู้ของเครื่อง (Machine Learning) วิธีหนึ่ง ซึ่งมักใช้ในปัญหาการแบ่งกลุ่ม (Classification) หรือปัญหาการจำแนกรูปแบบ (Pattern Classification) นิวรอลเน็ตเวิร์กประกอบด้วยนิวรอน (Neuron) จำนวนมากเชื่อมต่อกัน นิวรอนที่ใช้นั้นมีได้หลายรูปแบบ แบบที่นิยมใช้ได้แก่นิวรอนแบบซิกมอยด์ (Sigmoid unit) ซึ่งคำนวณผลลัพธ์จากอินพุตได้ตามสมการ (2.1) โดยกำหนดให้ $\mathbf{W}$ คือเวกเตอร์ของค่าน้ำหนัก (Weight) X คือเวกเตอร์ของค่าอินพุต และ σ คือฟังก์ชันกระตุ้น (Activation function) ประเภทฟังก์ชันซิกมอยด์ (Sigmoid function)

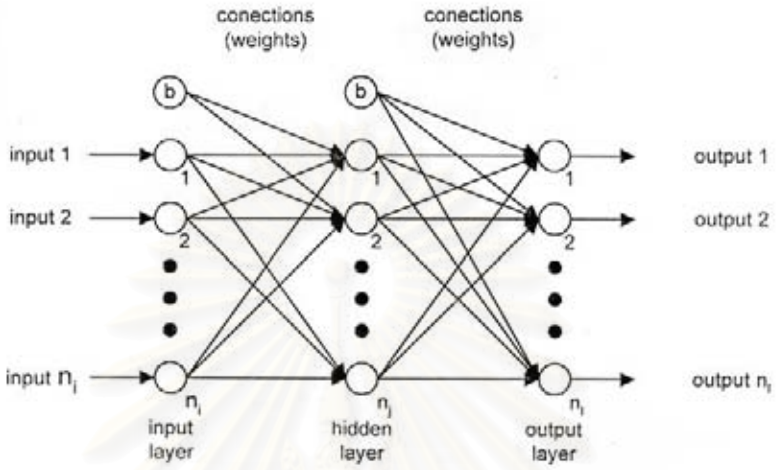
$$\mathbf{O} = \sigma(\mathbf{W} \cdot \mathbf{X}) \tag{2.1}$$

โดย

$$\sigma(y) = \frac{1}{1+e^{-y}} \tag{2.2}$$

รูปที่ 2.1 แสดงนิวรอลเน็ตเวิร์กที่มีโครงสร้าง 3 ชั้นคือชั้นอินพุต (Input layer) ชั้นซ่อน (Hidden layer) และชั้นเอาท์พุต (Output layer) แต่ละการเชื่อมต่อของนิวรอนมีค่าน้ำหนักกำหนดไว้ ค่าน้ำหนักเหล่านี้จะถูกปรับค่าให้เหมาะสมในระหว่างการเรียนรู้ จนกระทั่งได้ผลการเรียนรู้ถูกต้องที่สุดหรือมีความผิดพลาดน้อยที่สุด วิธีการเรียนรู้ของนิวรอลเน็ตเวิร์กแบบหนึ่งคือการเรียนรู้แบบแพร่กระจายย้อนกลับ (Backpropagation Learning Algorithm) ซึ่งทำโดยการผ่านข้อมูลฝึกสอนเข้าไปยังนิวรอลเน็ตเวิร์ก แล้วนำผลลัพธ์ไปเปรียบเทียบกับค่าเป้าหมาย (Target)

ผลต่างที่ได้คือค่าผิดพลาด ค่าผิดพลาดนี้จะนำไปใช้ในการคำนวณปริมาณการปรับค่าน้ำหนักเพื่อให้ค่าน้ำหนักของการเชื่อมต่อมีค่าที่เหมาะสม กระบวนการนี้จะทำซ้ำจนกระทั่งค่าผิดพลาดมีค่าน้อยในระดับที่ยอมรับได้หรือจนกระทั่งครบจำนวนรอบที่กำหนด การเรียนรู้แบบแพร่กระจายย้อนกลับสามารถอธิบายได้ดังรูปที่ 2.2 [2]



รูปที่ 2.1 นิวรอลเน็ตเวิร์กที่มี 3 ชั้น

X : vector of network input values

T : vector of target network output values

η : learning rate

x_{ji} : input from unit i to unit j

w_{ji} : weight from unit i to unit j

Create feed-forward network with n_{in} inputs, n_{hidden} hidden units, and n_{out} output units

Initialize all network weights to small random numbers

Until the termination condition is met, Do

For each training examples (*X, T*) Do

Input training instance *X* to the network and compute output *O*

For each network output unit *k*, calculate its error term δ_k

$$\delta_k \leftarrow O_k (1-O_k)(T_k-O_k)$$

For each hidden unit h, calculate its error term δ_h

$$\delta_h \leftarrow O_h (1-O_h)\Sigma_{k \in outputs} w_{kh} \delta_k$$

Update each network weight w_{ji}

$$w_{ji} \leftarrow w_{ji} + \Delta w_{ji} \quad , \quad \Delta w_{ji} = \eta \delta_k x_{ji}$$

รูปที่ 2.2 ขั้นตอนวิธีการเรียนรู้แบบแพร่กระจายย้อนกลับ

2.4 **แบบจำลองฮิดเด็นมาร์คอฟ** (Hidden Markov Model)

แบบจำลองฮิดเด็นมาร์คอฟเป็นวิธีการหนึ่งที่ใช้ได้กับปัญหาการจำแนกรูปแบบ โดยอาศัยวิธีการทางสถิติ แบบจำลองฮิดเด็นมาร์คอฟประกอบด้วย

1. สถานะ (State)

แบบจำลองประกอบมีจำนวนสถานะที่เป็นไปได้ N สถานะ โดยให้เซ็ตของสัญลักษณ์แทนสถานะต่างๆเป็น $S = \{ S_1 , S_2 , S_3 , ... , S_N \}$ และให้สถานะที่เวลา t เขียนแทนด้วย q_t

2. ค่าสังเกต (Observation)

แบบจำลองมีจำนวนค่าสังเกตที่เป็นไปได้เท่ากับ M โดยให้เซ็ตของสัญลักษณ์แทนค่าสังเกตต่างๆเป็น $V = \{ V_1 , V_2 , V_3 , ... , V_M \}$ และให้ $O = O_1O_2O_3...O_T$ แทนค่าสังเกตที่ได้ตั้งแต่เวลา t = 1 ถึง t = T

3. การแจกแจงความน่าจะเป็นในการเปลี่ยนสถานะ (State Transition Probability)

ให้ $A = \{ a_{ij} \}$ แทนเมตริกซ์การแจกแจงความน่าจะเป็นในการเปลี่ยนสถานะ โดย aij คือความน่าจะเป็นในการเปลี่ยนจากสถานะ S_i ไปเป็นสถานะ S_j นั่นคือ $a_{ij} = P[q_{t+1} = S_j \mid q_t = S_i]$, $1 \leq i , j \leq N$

4. การแจกแจงความน่าจะเป็นของค่าสังเกต (Observation Probability)

ให้ $B = \{ b_{jk} \}$ แทนเมตริกซ์การแจกแจงความน่าจะเป็นของค่าสังเกต โดย b_{jk} คือความน่าจะเป็นที่จะให้ค่าสังเกต V_k เมื่ออยู่ในสถานะ S_j ที่เวลา t นั่นคือ $b_{jk} = P[O_t = V_k \mid q_t = S_j]$, $1 \leq j \leq N , 1 \leq k \leq M$

5. การแจกแจงสถานะเริ่มต้น (Initial State Distribution)

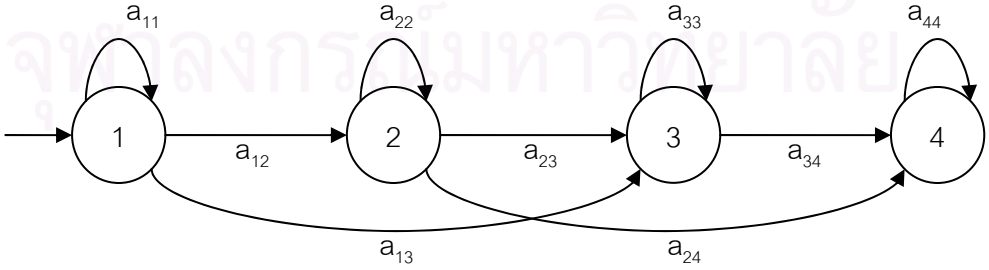
ให้ $\pi = \{ \pi_i \}$ แทนการแจกแจงสถานะเริ่มต้น โดย π_i คือความน่าจะเป็นที่สถานะเริ่มต้นคือ S_i นั่นคือ $\pi_i = P[q_1 = S_i]$, $1 \leq i \leq N$

จากที่ได้อธิบายไว้ข้างต้นจะเห็นได้ว่ารายละเอียดของแบบจำลองฮิดเด็นมาร์คอฟได้แก่ พารามิเตอร์ N และ M สัญลักษณ์ของสถานะและค่าสังเกต และการแจกแจงความน่าจะเป็น A, B, และ π จึงกำหนดสัญลักษณ์ $\lambda = (A, B, \pi)$ แทนเซ็ตของพารามิเตอร์ของแบบจำลอง

แบบจำลองฮิดเด็นมาร์คอฟนั้นถูกใช้งานในปัญหาพื้นฐาน 3 ประการได้แก่

1. ปัญหาที่หนึ่ง กำหนดให้ลำดับของค่าสังเกต $O = O_1O_2O_3...O_T$ และแบบจำลอง λ= (A, B, π) ให้หาความน่าจะเป็นของลำดับค่าสังเกต $P(O | \lambda)$ ซึ่งปัญหาแก้ได้โดยใช้ขั้นตอนวิธีไปข้างหน้า (Forward Algorithm) และกระบวนการย้อนกลับ (Backward Procedure) [3]
2. ปัญหาที่สอง กำหนดให้ลำดับของค่าสังเกต $O = O_1O_2O_3...O_T$ และแบบจำลอง λ= (A, B, π) ให้หาลำดับของสถานะที่ดีที่สุด ซึ่งสามารถอธิบายค่าสังเกตได้ ปัญหาที่สองนี้สามารถแก้ได้ด้วยขั้นตอนวิธีวิเทอร์บี (Viterbi Algorithm) [3]
3. ปัญหาที่สาม เป็นการปรับค่าพารามิเตอร์ λ= (A, B, π) เพื่อให้ได้ $P(O | \lambda)$ ที่มากที่สุด นั่นคือใช้ลำดับของค่าสังเกต O ในการฝึกสอน ขั้นตอนวิธี Baum-Welch [3] เป็นวิธีการหนึ่งที่ใช้สำหรับแก้ปัญหานี้

แบบจำลองฮิดเด็นมาร์คอฟมีหลายชนิดซึ่งแบ่งตามลักษณะโครงสร้างของแบบจำลอง ชนิดหนึ่งที่ใช้กันได้แก่ แบบจำลองฮิดเด็นมาร์คอฟแบบซ้ายไปขวา (Left-Right Hidden Markov Model) [3] ลักษณะเฉพาะของแบบจำลองชนิดนี้คือ เมื่อนำสถานะของแบบจำลองมาเรียงจากซ้ายไปขวา การเปลี่ยนสถานะจะเปลี่ยนจากสถานะทางซ้ายไปยังสถานะที่อยู่ทางขวาเท่านั้น ไม่มีการย้อนกลับจากขวามาซ้าย นอกจากนี้การทำงานจะเริ่มต้นที่สถานะที่อยู่ซ้ายสุดเท่านั้น รูปที่ 2.3 แสดงตัวอย่างของแบบจำลองซ้ายไปขวา ซึ่งมี 4 สถานะ กำหนดด้วยหมายเลข 1 ถึง 4 และให้ a_{ij} แทนความน่าจะเป็นในการเปลี่ยนสถานะจาก i ไป j จากรูปจะเห็นได้ว่าการทำงานจะเริ่มต้นที่สถานะที่ 1 และมีลักษณะแบบซ้ายไปขวาเช่น ที่สถานะที่ 3 จะย้ายไปยังสถานะที่ 4 (ด้วยความน่าจะเป็น a_{34}) หรือคงอยู่ที่สถานะที่ 3 (ด้วยความน่าจะเป็น a_{33}) ได้ แต่ไม่สามารถย้ายกลับไปยังสถานะที่ 1 และ 2 ได้ (นั่นคือ $a_{31} = a_{32} = 0$) นอกจากนี้แบบจำลองในรูปมีอันดับ (Order) ของการเปลี่ยนสถานะเป็น 2 นั่นคือสามารถย้ายสถานะเป็นระยะได้ไม่เกิน 2 สถานะ ดังเช่นสถานะที่ 1 จะย้ายไปที่สถานะที่ 2 หรือ 3 ได้เท่านั้น ไม่สามารถไปที่สถานะที่ 4 ได้

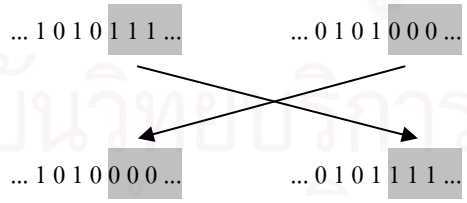


รูปที่ 2.3 การเปลี่ยนสถานะของแบบจำลองฮิดเด็นมาร์คอฟซ้ายไปขวา

2.5 **ขั้นตอนวิธีเชิงพันธุกรรม** (Genetic Algorithms)

ขั้นตอนวิธีเชิงพันธุกรรมอาศัยแนวความคิดของวิวัฒนาการทางธรรมชาติ (Natural Evolution) ในการค้นหาผลเฉลยที่ดีที่สุด โดยแทนผลเฉลยในรูปของโครโมโซม (Chromosome) แต่ละโครโมโซมมีค่าฟิตเนส (Fitness) เป็นตัวบอกคุณภาพของโครโมโซม(หรือผลเฉลยที่ถูกแทนด้วยโครโมโซมนั้น) และใช้ตัวดำเนินการทางพันธุกรรม (Genetic Operator) ได้แก่ การไขว้เปลี่ยน (Crossover) และการกลายพันธุ์ (Mutation) ในการสร้างโครโมโซมหรือคำตอบใหม่ ขั้นตอนวิธี จะพยายามรักษาโครโมโซมที่มีคุณภาพดีไว้ และพยายามลดโครโมโซมที่มีคุณภาพไม่ดีลง โดยให้โอกาสที่จะถูกเลือกสูงแก่โครโมโซมที่มีคุณภาพดี ในการที่จะขยายพันธุ์ (Reproduction) และอยู่รอดถึงในรุ่น (Generation) ต่อไป ในท้ายที่สุดแล้วโครโมโซมที่ดีที่สุดที่ได้ก็คือผลเฉลยที่ดีที่สุดที่เราต้องการหานั่นเอง ขั้นตอนวิธีเชิงพันธุกรรมอธิบายโดยย่อได้ดังนี้ [4]

1. เริ่มต้น สร้างประชากร (Population) ขึ้นมาอย่างสุ่มมีจำนวน n โครโมโซม
2. ประเมินค่าฟิตเนสของแต่ละโครโมโซมในประชากร
3. สร้างประชากรรุ่นใหม่ขึ้นมาด้วยวิธีดังนี้
   - 3.1. เลือกโครโมโซมพ่อแม่ (Parent Chromosome) มา 1 คู่ โดยเลือกด้วยความน่าจะเป็นตามค่าฟิตเนส
   - 3.2. ด้วยความน่าจะเป็นของการไขว้เปลี่ยน (Crossover Probability) ทำการไขว้เปลี่ยนให้เกิดโครโมโซมลูก (Offspring) ขึ้น รูปที่ 2.4 แสดงตัวอย่างการไขว้เปลี่ยน ส่วนที่แรเงาคือส่วนของโครโมโซมที่มีการไขว้เปลี่ยนกัน



รูปที่ 2.4 การไขว้เปลี่ยนของโครโมโซม

   - 3.3. ด้วยความน่าจะเป็นของการกลายพันธุ์ (Mutation Probability) ทำการกลายพันธุ์โครโมโซมลูก รูปที่ 2.5 แสดงตัวอย่างการกลายพันธุ์จากโครโมโซมเดิม (รูปบน) ไปเป็นโครโมโซมใหม่ (รูปล่าง) ส่วนที่แรเงาแสดงถึงส่วนที่มีการกลายพันธุ์
   - 3.4. ใส่โครโมโซมลูกลงในประชากร
4. ทำซ้ำข้อ 2, 3 จนกระทั่งถึงเงื่อนไขการสิ้นสุด เช่นจำนวนรุ่น หรือคุณภาพของโครโมโซม

5. คืนคำตอบเป็นโครโมโซมที่ดีที่สุดในประชากรรุ่นสุดท้าย

รูปที่ 2.5 การกลายพันธุ์

2.6 การวัดผลการค้นคืน

มาตรวัดที่ใช้วัดประสิทธิภาพของการค้นคืนได้แก่ ค่าแม่นยำ (Precision) ค่าเรียกคืน (Recall) [5] และตัววัด F1 (F1 Measurement) [6] ซึ่งมีวิธีคำนวณจากการนับจำนวนข้อมูลที่เกี่ยวข้อง (Relevant Data) และข้อมูลที่ระบบค้นคืนกลับมา (Retrieved Data) ได้ดังนี้

$$\text{ค่าแม่นยำ} = \frac{\text{จำนวนคำที่เกี่ยวข้องที่คืนกลับมา}}{\text{จำนวนคำที่คืนกลับมา}}$$

$$\text{ค่าเรียกคืน} = \frac{\text{จำนวนคำที่เกี่ยวข้อที่คืนกลับมา}}{\text{จำนวนคำที่คืนกลับมา}}$$

$$F1 = \frac{2\times\text{ค่าแม่นยำ}\times\text{ค่าเรียกคืน}}{\text{ค่าแม่นยำ}+\text{ค่าเรียกคืน}}$$

2.7 ขั้นตอนวิธีระยะแก้ไขสั้นที่สุด (Minimum Edit Distance)

ระยะแก้ไขสั้นที่สุด [7] เป็นเทคนิคหนึ่งในการวัดความคล้ายคลึงกันระหว่าง 2 สายอักขระ ซึ่งจะทำการคำนวณหาจำนวนคำสั่งน้อยที่สุดที่ใช้ในการเพิ่ม การลบ และการแทนที่แต่ละตัวอักขระ เพื่อทำให้สายอักขระทั้งสองสายเหมือนกัน ตัวอย่างเช่น ระยะห่างของการแก้ไขให้ EXSAMBL เป็น EXAMPLE เท่ากับ 3 ซึ่งมีวิธีการคำนวณดังนี้

ตัวอย่างที่ 2.1 การคำนวณระยะห่างของการแก้ไขให้ EXSAMBL เป็น EXAMPLE

1. การลบตัวอักษร S EXSAMBL → EXAMBL
2. การแทนที่ตัวอักษร B ด้วย P EXAMBL → EXAMPL
3. การเพิ่มตัวอักษร E EXAMPL → EXAMPLE

ดังนั้นระยะห่างของการแก้ไขให้ EXSAMBL เป็น EXAMPLE มีค่าเท่ากับ 3

จากวิธีการคำนวณข้างต้นสามารถเขียนในอยู่ในรูปการคำนวณด้วยความสัมพันธ์เวียนเกิด Edit (P_j, W_k) ได้ดังนี้

$$\begin{aligned}
&\text{Edit}(P_0, W_0) = 0\\
&\text{Edit}(P_j, W_0) = j\\
&\text{Edit}(P_0, W_k) = k\\
&\text{Edit}(P_j, W_k) = \min[\ \text{Edit}(P_{j-1}, W_k) + 1,\\
&\qquad\qquad \text{Edit}(P_j, W_{k-1}) + 1,\\
&\qquad\qquad \text{Edit}(P_{j-1}, W_{k-1}) + r(p_j, w_k)\ ]
\end{aligned}$$

โดยที่ $P_j = p_1 p_2 p_3 \ldots p_j$ เป็นสายอักขระต้นแบบ มีความยาว j ตัวอักษร

$W_k = w_1 w_2 w_3 \ldots w_k$ เป็นสายอักขระเป้าหมาย มีความยาว k ตัวอักษร

$r(p_j, w_k) = 0$ ถ้า p_j เท่ากับ w_k

1 ถ้า p_j ไม่เท่ากับ w_k

2.8 ขั้นตอนวิธีซาวด์เด็กซ์ภาษาอังกฤษ

M. K. Odell และ R. C. Russell ได้ออกแบบขั้นตอนวิธีการเข้ารหัสชื่อในภาษาอังกฤษ โดยยึดหลักของการอ่านออกเสียง เพื่อให้ชื่อที่อ่านออกเสียงคล้ายกันได้รหัสเหมือนกัน หรือที่เรียกว่า “ซาวด์เด็กซ์” (Soundex) ขั้นตอนวิธีดังกล่าวได้ใช้แนวคิดทางภาษาศาสตร์และชวเลขที่ว่าชื่อในภาษาอังกฤษสามารถจำแนกความแตกต่างได้โดยพิจารณาเพียงพยัญชนะเท่านั้น [8]

รูปที่ 2.6 แสดงขั้นตอนการเข้ารหัสซาวด์เด็กซ์ โดยเริ่มจากการนำตัวอักษรตัวแรกของคำไปเป็นรหัส ส่วนตัวอักษรที่เหลือจะแปลงเป็นตัวเลขโดยใช้ตารางการกำหนดรหัสซาวด์เด็กซ์ ดังแสดงในตารางที่ 2.1 จากนั้นจะตัดรหัสตัวเลขศูนย์ออกไป และถ้ารหัสตัวเลขที่อยู่ตำแหน่งติดกันมีค่าเท่ากันจะเก็บเพียงหนึ่งรหัสเท่านั้น สุดท้ายรหัสซาวด์เด็กซ์ที่ได้คือตัวอักษรตัวแรกของชื่อตามด้วยรหัสตัวเลขสามตัวแรกที่ได้จากการแปลง ถ้ารหัสที่ได้มีความยาวไม่ถึงสี่หลักจะเติมตัวเลขศูนย์จนครบสี่หลัก ตัวอย่างเช่น คำว่า “ALEXANDER” มีรหัสซาวด์เด็กซ์เท่ากับ A425 เป็นต้น

```
char *SOUNDEX(char *Name)
{
                        /*ABCDEFGHIJKLMNOPQRSTUVWXYZ*/
    char Table[] = “01230120022455012623010202”;
    char Code[] =  “0000”;
    int    Count = 0;
    char Ch;
    /*----- For the First Character -----*/
    Code[Count++] = Name[0];
    /*----- For the Rest Character -----*/
    for (i=2; i  < strlen (Name); i++)  {
        Ch = Table[Name[i] - 'A'];
        if  (Ch != PrevCode && Ch != '0')  {
            Code[Count++] := Ch;
            if  (Count = 5)
                return(Code);
        }
        PrevCode := Ch;
    }
    return(Code);
}
```

รูปที่ 2.6 โปรแกรมการเข้ารหัสซาวด์เด็กซ์ภาษาอังกฤษ

2.9 **งานวิจัยของ วรรณี อุดมพาณิชย์**

งานวิจัย [8] เสนอกฎเกณฑ์การสร้างรหัสคำได้แก่ (1) ไม่ใช้สระ วรรณยุกต์ และไม้ไต่คู้มาสร้างรหัส ยกเว้นสระที่ให้เสียงตัวสะกดเป็นพยัญชนะ ได้แก่ ไ- ใ- –ำ (2) เปลี่ยน ไ- ใ- ไ-ย -ย เป็น -ย (3) เปลี่ยน รร เป็น –ัน (4) ตัดการันต์ พยัญชนะที่มีการันต์กำกับ รวมทั้งสระและอักษรที่ควบการันต์ทิ้ง (5) ใช้รหัสความยาว 7 หลัก รหัสตัวแรกเป็นตัวอักษรซึ่งใช้ตารางเทียบรหัสดังแสดงในตารางที่ 2.2 ส่วนรหัสตัวที่เหลือเป็นตัวเลขซึ่งใช้ตารางที่ 2.3 ในการเทียบรหัส ถ้ารหัสที่ได้

ตารางที่ 2.1 การกำหนดรหัสซาวด์เด็กซ์ภาษาอังกฤษของ Odell และ Russel

ตัวอักษร	รหัสตัวเลข
A E I O U H W Y	0
B F P V	1
C G J K Q S X Z	2
D T	3
L	4
M N	5
R	6

ตารางที่ 2.2 การกำหนดรหัสตัวอักษรของรหัสซาวด์เด็กซ์ภาษาไทย
จากงานวิจัยของวรรณี อุดมพาณิชย์

รหัสตัวอักษร	ตัวอักษร
ก	ก
ข	ข ฃ ค ฅ ฆ
ง	ง
จ	จ
ช	ช ฌ ฉ
ส	ซ ศ ษ ส
ด	ด ฎ
ต	ต ฏ
ท	ฐ ฑ ฒ ถ ท ธ
น	ณ น

รหัสตัวอักษร	ตัวอักษร
บ	บ
ป	ป
พ	พ ภ ผ
ฟ	ฝ ฟ
ม	ม
ย	ญ ย
ร	ร ล ฬ ฤ ฦ
ว	ว
อ	อ
ฮ	ห ฮ

ตารางที่ 2.3 การกำหนดรหัสตัวเลขของรหัสซาวด์เด็กซ์ภาษาไทย
จากงานวิจัยของของวรรณี อุดมพาณิชย์

รหัสตัวเลข	ตัวอักษร
0	ม ว –ำ
1	ก ข ฃ ค ฅ ฆ
2	ง ย
3	ญ ณ น
4	ฎ ฏ ด ต ศ ษ ส
5	บ ป พ ภ
6	ผ ฝ ฟ ห อ ฮ
7	จ ฉ ช ซ ฌ
8	ฐ ฑ ฒ ถ ท ธ
9	ร ฤ ล ฦ ฬ

มีความยาวน้อยกว่า 7 หลัก ให้เติม 0 จนครบ นอกจากนี้ยังได้เพิ่มกฎเกณฑ์เพื่อให้เหมาะสมกับภาษาไทย ได้แก่

- กรณีพบ ใ- ไ- ไ-ย และ –ัย จะเปลี่ยนให้อยู่ในรูปแบบเดียวกันคือ –ัย ก่อนทำการเข้ารหัส เนื่องจากสระดังกล่าวอ่านออกเสียงเหมือนกัน เช่น ใท ไท ไทย และ ทัย จะเปลี่ยนเป็น ทัย
- กรณีพบ รร จะเปลี่ยนเป็น –ัน ในกรณีที่ไม่มีตัวสะกดตามหลัง และเปลี่ยนเป็น –ั กรณีที่มีตัวสะกดตามหลัง เช่น เปลี่ยนคำว่า สรรเพชร รังสรรค์ พรรณนที และ ธรรมรัตน์ เป็น สันเพชร รังสัน พัณนที และ ธัมรัตน์ ตามลำดับ
- กรณีพบการันต์ จะตัดการันต์และพยัญชนะที่มีตัวการันต์กำกับรวมทั้งสระและอักษรควบการันต์ทิ้ง เช่น คำว่า จันทร์ ศักดิ์ และ พันธุ์ เปลี่ยนเป็น จัน ศัก และ พัน ตามลำดับ

ตัวอย่างการเข้ารหัสคำ เช่น คำ “อัมพร” หรือ “อำภรณ์” จะได้รหัสเป็น “อ059000” คำ “พรรณศักดิ์” หรือ “พันธุ์ศักดิ์” จะได้รหัสเป็น “พ341000” คำ “เนืองนิตย์” หรือ “เนืองนิจ” จะได้รหัสเป็น “น623400”

2.10 งานวิจัยของ นิลเนตร อรุณวงศ์ ณ อยุธยา

งานวิจัย [10] มีการปรับวิธีการจาก [9] ได้แก่ (1) ใช้อักขระไทยทั้งหมดในการเข้ารหัส โดยใช้ตารางที่ 2.4 เป็นตารางกำหนดรหัสสำหรับอักขระตัวแรกของรหัส และตารางที่ 2.5 สำหรับอักขระตัวที่เหลือของรหัส (2) ตัดตัวควบกล้ำทิ้ง เช่น ปรับ เป็น ปับ หรือ คลอง เป็น คอง (3) อักษรนำเสียงสนิท (ได้แก่ อย, หง, หญ, หน, หม, หย, หร, หล, หว ให้ตัด “อ“ หรือ “ห“ ทิ้ง (4) เปลี่ยนตำแหน่งสระหน้า (เช่น เ- แ- โ- ไ- ใ-) ไปไว้หลังสุด (5) ตัวอักษรติดกันและเหมือนกันให้ยุบเหลือรหัสตัวเดียว (6) ไม่นำสระ -ะ และ ิ- มาเข้ารหัส ตัวอย่างการเข้ารหัสคำ เช่น “กิตติพรรณ” แทนด้วย “กตบน” ซึ่งได้จากการตัด ิ , ตัว “ต” ติดกัน 2 ตัว ตัดเหลือตัวเดียว, ตัว “พ” แทนด้วยรหัส “บ”, แทน “รร” ด้วย ั้น แล้วตัดตัว ั ออก

ตัวอย่างการเข้ารหัสคำ เช่น คำ “อัมพร” หรือ “อำภรณ์” จะได้รหัสเป็น “อมบน” คำ “พรรณศักดิ์” หรือ “พันธุ์ศักดิ์” จะได้รหัสเป็น “พนดก” คำ “เนืองนิตย์” หรือ “เนืองนิจ” จะได้รหัสเป็น “นืองนดเ”

ตารางที่ 2.4 การกำหนดรหัสสำหรับอักขระตัวแรก
จากงานวิจัยของนิลเนตร อรุณวงศ์ ณ อยุธยา

รหัสตัวอักษร	ตัวอักษร	รหัสตัวอักษร	ตัวอักษร
ก	ก	บ	บ
ข	ข ฃ ค ฅ ฆ	ป	ป
ง	ง	พ	พ ภ ผ
จ	จ	ฟ	ฝ ฟ
ช	ช ฌ ฉ	ม	ม
ซ	ซ ศ ษ ส	ย	ญ ย
ด	ด ฎ	ร	ร ล ฬ ฤ ฦ
ต	ต ฏ	ว	ว
ท	ฐ ฑ ฒ ถ ท ธ	อ	อ
น	ณ น	ฮ	ห ฮ

ตารางที่ 2.5 การกำหนดรหัสสำหรับอักขระถัดจากตัวแรก
จากงานวิจัยของนิลเนตร อรุณวงศ์ ณ อยุธยา

รหัสตัวอักษร	ตัวอักษร
ก	ก ข ค ฆ
ง	ง
ด	จ ฉ ช ซ ฌ ฎ ฏ ฐ ฑ ฒ ด ต ถ ท ธ ศ ส ษ
น	ญ ณ น ร ล ฬ
บ	บ ป พ ฟ ภ ผ ฝ
ม	ม ำ
ย	ย
ว	ว
ไ	ไ-ย -ย ใ-

2.11 งานวิจัยของ ประยุทธ สุวรรณวิสาท และ สมชาย ประสิทธิ์จูตระกูล

งานวิจัยนี้ออกเป็น 2 ส่วนหลักคือ การค้นคืนข้ามภาษาไทยทับศัพท์ภาษาอังกฤษ และการค้นคืนข้ามภาษาอังกฤษทับศัพท์ภาษาไทย โดยที่กรณีการค้นคืนข้ามภาษาไทยทับศัพท์ภาษาอังกฤษ [11] นั้นใช้รหัสคำเป็นตัวเลขทั้งหมดโดยไม่จำกัดความยาวของรหัสคำ ตารางที่ 2.6 แสดงการกำหนดรหัสสำหรับคำไทยทับศัพท์คำอังกฤษ สำหรับการค้นคืนนั้นใช้การเปรียบเทียบรหัสคำแบบเหมือนกันทุกประการ นั่นคือรหัสคำต้องตรงกันทุกตัวจึงจะถือว่าสองคำนั้นมีเสียงอ่านตรงกัน จากผลการทดลองพบว่าเมื่อใช้รหัสคำความยาวมากกว่า 4 หลักขึ้นไป ได้ค่าแม่นยำ 78% และค่าเรียกคืน 90%

สำหรับกรณีการค้นคืนข้ามภาษาอังกฤษทับศัพท์ภาษาไทย [12] ใช้รหัสคำเป็นตัวอักษรไทยผสมกับสัญลักษณ์เสียงสากล ตารางที่ 2.7 และตารางที่ 2.8 แสดงรหัสสำหรับพยัญชนะและสระตามลำดับ สำหรับคำไทยมีการประมวลผลเบื้องต้นก่อนการเข้ารหัสคำ ซึ่งได้แก่ การลดรูป การตัดวรรณยุกต์และตัวการันต์ การแทนที่ ไ- ใ- ไ-ย -ย ด้วย ั-ย แทนที่ รร เป็น ั-น เป็นต้น ในการเปรียบเทียบรหัสคำนั้นใช้วิธีการเปรียบเทียบเชิงประมาณด้วยเทคนิคระยะแก้ไขสั้นสุด และมีการกำหนดต้นทุนในการแทนที่อักขระสำหรับแต่ละคู่อักขระตามกฎเกณฑ์ที่ได้สร้างขึ้น ค่าของต้นทุนมี 4 ระดับคือ C_1 C_2 C_3 และ C_4 ซึ่งมีค่า 0 1 4 และ 7 ตามลำดับ ดังตัวอย่างในรูปที่ 2.7 และรูปที่ 2.8 จากผลการทดลองพบว่าได้ค่าแม่นยำ 69% และค่าเรียกคืน 73%

ตารางที่ 2.6 การกำหนดรหัสสำหรับคำอังกฤษและคำไทยทับศัพท์คำอังกฤษ จากงานวิจัยของ ประยุทธ สุวรรณวิสาทร

ภาษาอังกฤษ	ภาษาไทย	รหัส
A E I O U H W Y [2]	อ ห ฮ ย ญ	0
B F P V	บ ฝ ฟ ป ผ พ ภ ว	1
C G J K Q S X Z	ข ฃ ค ฅ ฆ ฉ ช ฌ ก จ ฐ ศ ษ ส	2
D T	ฎ ด ฏ ต ฐ ฑ ฒ ถ ท ธ	3
L	ล ฬ	4
M N	ม ณ น	5
R	ร	6
A E I O U [1]	อ	7
H [1]	ห ฮ	8
W [1]	ว	1
Y [1]	ย ญ	9
	ง	52

1 : สำหรับตัวอักษรแรกของคำ

2 : สำหรับตัวอักษรตั้งแต่ตัวที่ 2 เป็นต้นไปของคำ

77% สำหรับกรณีคำไทยทับศัพท์คำอังกฤษ และได้ค่าแม่นยำ 96% และค่าเรียกคืน 75% สำหรับกรณีคำอังกฤษทับศัพท์คำไทย

2.12 งานวิจัยของ ทัศนวรรณ ศูนย์กลาง สมชาย ประสิทธิ์จูตระกูล และบุญเสริม กิจศิริกุล

งานวิจัย [13] เสนอการเข้ารหัสคำด้วยนิวรอลเน็ตเวิร์กแบบแพร่กระจายย้อนกลับ โดยการพิจารณาตัวอักษรที่อยู่ข้างเคียงกับตัวที่กำลังพิจารณา ในงานวิจัยนี้ใช้ตัวอักษรที่อยู่ข้างเคียงข้างหน้าและข้างหลังอย่างละ 4 ตัว ดังนั้นเมื่อรวมตัวอักษรที่กำลังพิจารณาแล้ว ข้อมูลเข้าของนิวรอลเน็ตเวิร์กจึงมี 9 ตัวอักษร ส่วนข้อมูลออกคือรหัสเสียงจากข้อมูลขาเข้า ดังแสดงในรูปที่ 2.9 และรูปที่ 2.10 นอกจากนี้การเข้ารหัสคำของคำไทยยังมีการใช้การประมวลผลเบื้องต้นด้วยเช่นเดียวกับในงานวิจัย [12] ในการค้นคืนใช้วิธีการเปรียบเทียบรหัสคำแบบประมาณด้วยวิธีระยะแก้ไขสั้น

ตารางที่ 2.7 การกำหนดรหัสของพยัญชนะสำหรับคำไทยและคำอังกฤษทับศัพท์คำไทย จากงานวิจัยของ ประยุทธ สุวรรณวิสาทร

อักษรอังกฤษ	อักษรไทย	รหัส
b	บ	บ
bh	พ	พ
c	ซ	ซ
ch	ช ฉ ฌ	ช
ck	ก	ก
d	ด ฎ	ด
dh	ท	ท
f	ฟ ฝ	ฟ
g	ก	ก
h	ห ฮ	ห
j	จ	จ
k	ก	ก
kh	ข ฃ ค ฅ ฆ	ข
l	ล ฦ ฬ	ล
m	ม	ม

อักษรอังกฤษ	อักษรไทย	รหัส
n	น ณ	น
ng	ง	ง
p	ป	ป
ph	พ ผ ภ	พ
q	ค	ค
r	ร ฤ	ร
s	ส ซ ศ ษ	ส
t	ต ฏ	ต
th	ท ฐ ฑ ฒ ถ ธ	ท
v	ว	ว
w	ว	ว
x	ก	ก
y	ย ญ	ย
z	ซ	ซ

ที่สุด โดยยอมให้ความต่างของรหัสคำมีได้ไม่เกิน 1 และกำหนดค่าต้นทุนการแก้ไขอักขระทั้งการเพิ่ม ลบ และแทนที่ ให้มีค่าเท่ากับ 1 จากผลการทดลองพบว่าได้ค่าแม่นยำ 87% และค่าเรียกคืน

2.13 **งานวิจัยของ** T. Duangpanyasawang and B. Kijsirikul

งานวิจัย [14] เสนอการใช้แบบจำลองฮิดเด็นมาร์คอฟร่วมกับไตรแกรมทางเสียง (Phonetic Tri-Grams) เพื่อการเข้ารหัสคำ ได้กำหนดให้สถานะของแบบจำลองฮิดเด็นมาร์คอฟแทนรหัสเสียงในระดับพยางค์ซึ่งประกอบด้วย 3 ส่วนคือรหัสพยัญชนะต้น รหัสสระ และรหัสตัวสะกด ส่วนค่าสังเกตเป็นรูปแบบของพยางค์ที่สอดคล้องกับสถานะนั้น และให้นิยามความน่าจะเป็นในการเปลี่ยนสถานะคือความน่าจะเป็นที่รหัสเสียงสองรหัสจะอยู่ติดกัน รูปที่ 2.11 แสดง

ตารางที่ 2.8 การกำหนดรหัสของสระสำหรับคำไทยและคำอังกฤษทับศัพท์คำไทย
จากงานวิจัยของ ประยุทธ สุวรรณวิสาทร

ตัวอักษรอังกฤษ	ตัวอักษรไทย	รหัส
-a	ะ	ะ
-aa	า	า
-ae	แ-ะ แ-	x
-ai	◌ัย	◌ัย
-ao	เ-า	@
-aiu	เ-ีย	I
-arn	าน	าน
-art	าท	าท
-e	เ-ะ เ-	เ
-ee	◌ี	◌ี
-eo	แ-ว	แ-ว
-er	เ-อ เ◌ิ	q

ตัวอักษรอังกฤษ	ตัวอักษรไทย	รหัส
-eu	◌ึ	◌ึ
-i	◌ิ	◌ิ
-ia	เ-ียะ เ-ีย	I
-ie	เ-ียะ เ-ีย	I
-o	-อ	อ
-oe	เ-อ เ-	Q
-oi	อย	อย
-oo	◌ู	◌ู
-orn	-อน	ร
-u	◌ึ ◌ื ◌ุ ◌ู	◌ุ
-ua	เ-ือะ เ-ือ -ั วะ ◌ัว	U
-ue	◌ื	◌ื

	ก	ข	ค	…	ท	ธ	น	บ	…
ก	C_1	C_2	C_2	…	C_4	C_4	C_4	C_4	…
ข	C_2	C_1	C_1	…	C_4	C_4	C_4	C_4	…
ค	C_2	C_1	C_1	…	C_4	C_4	C_4	C_4	…
…	…	…	…	…	…	…	…	…	…
ท	C_4	C_4	C_4	…	C_1	C_1	C_4	C_4	…
ธ	C_4	C_4	C_4	…	C_1	C_1	C_4	C_4	…
น	C_4	C_4	C_4	…	C_4	C_4	C_1	C_4	…
บ	C_4	C_4	C_4	…	C_4	C_4	C_4	C_1	…
…	…	…	…	…	…	…	…	…	…

รูปที่ 2.7 ตัวอย่างการกำหนดต้นทุนการแทนที่อักขระสำหรับพยัญชนะ
จากงานวิจัยของประยุทธ สุวรรณวิสาทร

	ะ	ั	า	ิ	ี	ึ	ื	ุ	...
ะ	C_1	C_1	C_1	C_4	C_4	C_4	C_4	C_4	...
ั	C_1	C_1	C_2	C_4	C_4	C_4	C_4	C_2	...
า	C_1	C_2	C_1	C_4	C_4	C_4	C_4	C_4	...
ิ	C_4	C_4	C_4	C_1	C_1	C_3	C_4	C_4	...
ี	C_4	C_4	C_4	C_1	C_1	C_4	C_4	C_4	...
ึ	C_4	C_4	C_4	C_3	C_4	C_1	C_1	C_2	...
ื	C_4	C_4	C_4	C_4	C_4	C_1	C_1	C_2	...
ุ	C_4	C_2	C_4	C_4	C_4	C_2	C_2	C_1	...
...	...	...	...	...	...	...	...	...	...

รูปที่ 2.8 ตัวอย่างการกำหนดต้นทุนการแทนที่อักขระสำหรับสระ
จากงานวิจัยของประยุทธ สุวรรณวิสาทร

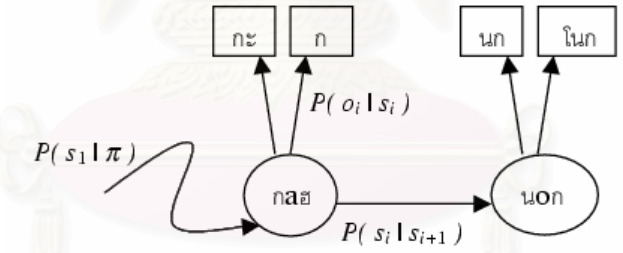
ลำดับตัวอักษร	รหัส
_ , _ , _ , _ , ก , น , ก , พ , ี	k
_ , _ , _ , ก , น , ก , พ , ี , ร	a, n
_ , _ , ก , น , ก , พ , ี , ร , ะ	o, k
_ , ก , น , ก , พ , ี , ร , ะ , ว	p
ก , น , ก , พ , ี , ร , ะ , ว , ุ	i
น , ก , พ , ี , ร , ะ , ว , ุ , ฒ	r
ก , พ , ี , ร , ะ , ว , ุ , ฒ , ิ	a
พ , ี , ร , ะ , ว , ุ , ฒ , ิ , _	v
ี , ร , ะ , ว , ุ , ฒ , ิ , _ , _	u
ร , ะ , ว , ุ , ฒ , ิ , _ , _ , _	t
ะ , ว , ุ , ฒ , ิ , _ , _ , _ , _	_

รูปที่ 2.9 ตัวอย่างการเข้ารหัสคำไทยของทัศนวรรณ ศูนย์กลาง และคณะ

ลำดับตัวอักษร	รหัส
_ , _ , _ , _ , R , H , O , D , I	r
_ , _ , _ , R , H , O , D , I , U	_
_ , _ , R , H , O , D , I , U , M	o
_ , R , H , O , D , I , U , M , _	d
R , H , O , D , I , U , M , _ , _	I
H , O , D , I , U , M , _ , _ , _	_
O , D , I , U , M , _ , _ , _ , _	m

รูปที่ 2.10 ตัวอย่างการเข้ารหัสคำอังกฤษของทัศนวรรณ ศูนย์กลาง และคณะ

ตัวอย่างส่วนหนึ่งแบบจำลองฮิดเด็นมาร์คอฟ จากรูป สถานะ “กaฮ” แทนการออกเสียง “กะ” ซึ่งค่าสังเกตที่เป็นไปได้สำหรับสถานะนี้คือ “กะ” และ “ก” ส่วนสถานะ “นoก” แทนการออกเสียง “นก” ซึ่งค่าสังเกตที่เป็นไปได้คือ “นก” และ “โนก”



รูปที่ 2.11 แบบจำลองฮิดเด็นมาร์คอฟและไตรแกรมการออกเสียงในการเข้ารหัสคำ

ในการเข้ารหัสคำนั้นจะนำคำมาผ่านการประมวลผลเบื้องต้น แล้วสร้างเป็นลำดับของแบบพยางค์ที่เป็นไปได้ทั้งหมดออกมา จากนั้นใช้แบบจำลองฮิดเด็นมาร์คอฟเพื่อหารหัสคำของแต่ละลำดับพยางค์ จากนั้นเลือกรหัสคำที่ดีที่สุดจากการคำนวณค่าความน่าจะเป็นร่วม (Combined probability) CP จากแบบจำลองฮิดเด็นมาร์คอฟและไตรแกรมทางเสียง ซึ่งคำนวณได้ดังนี้

กำหนดให้

- แบบจำลองฮิดเด็นมาร์คอฟ H = {π, A, B}
- ลำดับของพยางค์อยู่ในรูป X = $o_1o_2\ldots o_m$ โดย o_1, o_2, …, o_m $\in$ O ซึ่งเป็นเซ็ตของค่าสังเกต

ลำดับสถานะ S = $s_1 s_2 \ldots s_m$ ซึ่งให้ค่าความน่าจะเป็น P(S | X) มากที่สุด คำนวณจากแบบจำลองฮิดเด็นมาร์คอฟได้ดังนี้

$$P_{HMM}(S \mid X) = P(s_1 \mid \pi)\prod_i P(s_i \mid s_{i+1})P(o_i \mid s_i) \tag{2.1}$$

และคำนวณความน่าจะเป็นของลำดับสถานะ S = $s_1 s_2 \ldots s_m$ ด้วยไตรแกรมได้ดังนี้

$$P_{Trigram}(S) = \prod_i P(s_i \mid s_{i+1}, s_{i+2}) \tag{2.2}$$

ดังนั้นความน่าจะเป็นร่วม CP (Combined probability) จากแบบจำลองฮิดเด็นมาร์คอฟและไตรแกรมทางเสียง คำนวณได้โดยใช้สมการ (2.1) และ (2.2)

$$CP = P(s_1 \mid \pi)\prod_i P(s_i \mid s_{i+1})P(o_i \mid s_i) \cdot \prod_i kP(s_i \mid s_{i+1}, s_{i+2}) \tag{2.3}$$

โดย k คือค่าคงที่ซึ่งแทนความสัมพันธ์ระหว่างความน่าจะเป็นจากแบบจำลองฮิดเด็นมาร์คอฟและความน่าจะเป็นจากไตรแกรม

จากผลการทดลองพบว่าเมื่อใช้แบบจำลองฮิดเด็นมาร์คอฟร่วมกับไตรแกรมทางเสียง ได้ค่าแม่นยำ 95% และค่าเรียกคืน 84% ซึ่งได้ผลดีกว่าเมื่อใช้แบบจำลองฮิดเด็นมาร์คอฟอย่างเดียว ซึ่งได้ค่าแม่นยำ 90% และค่าเรียกคืน 83%

2.14 สรุป

ในบทนี้ได้กล่าวถึงทฤษฎีที่เกี่ยวข้องกับงานวิจัยนี้ ได้แก่ การถอดอักษร การถ่ายเสียงนิวรอลเน็ตเวิร์ก แบบจำลองฮิดเด็นมาร์คอฟ ขั้นตอนวิธีเชิงพันธุกรรม การวัดผลการค้นคืน และขั้นตอนวิธีระยะแก้ไขสั้นสุด จากนั้นได้กล่าวถึงงานวิจัยต่างๆที่เกี่ยวข้องกับการเข้ารหัสคำ และการค้นคืนข้ามภาษาไทย-อังกฤษ ในบทต่อๆไปจะกล่าวถึงการนำทฤษฎีและงานวิจัยที่เกี่ยวข้องมาใช้ในงานวิจัยนี้

บทที่ 3

การเข้ารหัสคำ

ในการค้นคืนข้ามภาษาโดยไม่อาศัยพจนานุกรมนั้น จำเป็นจะต้องอาศัยการใช้รหัสคำ โดยการนำคำสำคัญต่างๆในฐานข้อมูลมาสร้างรหัสคำเก็บไว้ในดัชนี และเมื่อมีการสืบค้นข้อมูลก็สร้างรหัสคำของคำสำคัญที่ต้องการค้น แล้วนำรหัสคำที่ได้นั้นไปค้นในดัชนีรหัสคำที่ได้สร้างไว้ ในบทนี้จะกล่าวถึงวิธีการเข้ารหัสคำ ได้แก่ การกำหนดรหัสเสียงในภาษาไทยและภาษาอังกฤษ การประมวลผลเบื้องต้นสำหรับคำไทย การเข้ารหัสคำด้วยนิวรอลเน็ตเวิร์กและแบบจำลองฮิดเด็นมาร์คอฟ

3.1 รหัสคำ

รหัสคำคือสัญลักษณ์แทนเสียงอ่านของคำ ดังนั้นคำที่มีเสียงอ่านตรงกันจะมีรหัสคำที่เหมือนกันหรือใกล้เคียงกัน รหัสคำประกอบด้วยรหัสเสียงซึ่งแทนเสียงของตัวอักขระแต่ละตัวที่อยู่ในคำนั้นมาเรียงต่อกัน ในงานวิจัยนี้ใช้หลักเกณฑ์การออกเสียงทั้งภาษาไทยและภาษาอังกฤษของราชบัณฑิตยสถาน [15] [16] มาสร้างเป็นตารางรหัสเสียงดังแสดงในตารางที่ 3.1 และตารางที่ 3.2 โดยในกรณีคำไทยทับศัพท์คำอังกฤษจะใช้หน่วยเสียงภาษาอังกฤษเป็นหลัก แล้วนำหน่วยเสียงภาษาไทยไปเทียบเพื่อหากลุ่มเสียงที่ตรงกันหรือใกล้เคียงกัน ในทางกลับกันในกรณีคำอังกฤษทับศัพท์คำไทยจะใช้หน่วยเสียงภาษาไทยเป็นหลัก แล้วนำหน่วยเสียงภาษาอังกฤษไปเทียบ

3.2 การประมวลผลเบื้องต้น

ในกรณีของคำภาษาไทยนั้น ต้องนำคำที่ต้องการเข้ารหัสมาผ่านการประมวลผลเบื้องต้นก่อนเพื่อลดความซับซ้อนในการเข้ารหัสคำ รายละเอียดยึดหลักตาม [9] ดังนี้

1. ตัดวรรณยุกต์และไม้ไต่คู้
2. เปลี่ยน รร เป็น –ัน หรือ –ั
3. เปลี่ยน ไ- ใ- ไ-ย เป็น -ัย
4. เปลี่ยน –ำ เป็น –ัม
5. ตัดตัวการันต์ และอักษรควบตัวการันต์

ตารางที่ 3.1 รหัสเสียงสำหรับคำไทยทับศัพท์คำอังกฤษ

เสียงพยัญชนะ		รหัสเสียง	เสียงสระ		รหัสเสียง
ไทย	อังกฤษ		ไทย	อังกฤษ	
พ	p	p	◌ิ ◌ี	ee, ei, ea, ey, i	i
บ	b	b	เ	e, ay	e
ท,ต	t, th	t	แ	a, air, are	w
ด	d, th	d	เ◌ียว	a, aw, au	@
ก,ค	c, k, g	k	◌ุ , ◌ู	u oo	u
ช	ch, sh	c	เ◌อ, เ◌ิ	ur, er, ir, or	W
จ	j, ch, g	j	◌ะ, ◌ั	a	a
ฟ	f, ph	f	โ	ome, o	o
ว	w, v	v	ไ, ใ, ◌ัย, ◌าย	ie, ai	!
ส,ซ	s, z	s	◌าว, เ◌า	ow, ou, our, au	R
ฮ	h	h	◌วย	oi	O
ม	m	m	เ◌ีย	ear, ia	I
น	n	n	◌ัว	our, ua	Y
ง	ng	g	◌ิว	ew, eua	X
ล	l	l	เ◌ิล	le	Q
ร	r	r			
ย	y	y			
ตัวอักษรที่ไม่ออกเสียง		_			

ตารางที่ 3.2 รหัสเสียงพยัญชนะสำหรับคำอังกฤษทับศัพท์คำไทย

เสียงพยัญชนะ		รหัสเสียง		เสียงสระ		รหัสเสียง
ไทย	อังกฤษ	ตัวต้น	ตัวสะกด	ไทย	อังกฤษ	
ก ข ค ฆ	ck, g, k, x, c, kh, q	k	k	◌ิ ◌ี	i, ee	i
ง	ng	g	g	ะ า ◌ั	a, u, ar	a
จ ฉ ช ฌ	j, ch, x	c	t	เ–ะ เ–	e	e
ซ ส ศ ษ สร ทร	s, z	s	t	แ–ะ แ–	ae	w
ญ ย หย หญ	y	y	n	◌ึ ◌ื ◌ือ ◌ุ ◌ู	u, eu, ue, eo, oo	u
ด ฎ ฑ	d	d	t	โ–ะ โ– เ–าะ –อ	o	o
ต ฏ ถ ฐ ท ฑ ธ ฒ	t, th, dh	t	t	เ–อะ เ–อ เ◌็	er, oe	W
ณ น ห น	n	n	n	เ◌ียะ เ◌ีย	ia, ie, aiu	I
บ	b	b	p	เ◌ือะ เ◌ือ ◌ัวะ ◌ัว	ua, ue, ea, ui	Y
ป ผ พ ภ	p, ph, bh	p	p	ไ– ใ– ไ–ย ◌ัย –าย	ai, ie, uy	!
ฝ ฟ	f	f	p	เ–า –าว	ao, ou, ow	R
ม	m	m	m	โ–ย –อย	oi, oy	x
ร ฤ	r	r	n	◌ิว	iu	X
ล ฬ ฦ	l	l	n	เ–ว	eo	q
ว	w, v	v	–	เ-ย	oei	Q
ห ฮ	h	h	–	เ◌ือย –วย	uai, uay, ou	O
ตัวอักษรที่ไม่ออกเสียง	–	–		แ-ว	aeo, eo, aew	$

ตารางที่ 3.2 (ต่อ) รหัสเสียงพยัญชนะสำหรับคำอังกฤษทับศัพท์คำไทย

เสียงพยัญชนะ		รหัสเสียง		เสียงสระ		รหัสเสียง
ไทย	อังกฤษ	ตัวต้น	ตัวสะกด	ไทย	อังกฤษ	
				เ-ียว	ieo, eaw, eo, ew, iow, iau, iew, iaw	@

6. เปลี่ยน ฤ เป็น รึ ริ หรือ เรอ และเปลี่ยน ฤๅ เป็น รือ โดยตัว ฤ ต้องพิจารณาดังนี้ [17]
   6.1. ฤ ออกเสียงเป็น เรอ มีคำเดียว คือ ฤกษ์ โดยเปลี่ยนเป็น เริก
   6.2. ฤ ออกเสียงเป็น ริ ถ้าประสมกับ ก ต ท ป ศ ส เช่น กฤษณา ตฤณ ทฤษฎี ปฤงคพ ศฤงคาร สฤษฏ์
   6.3. ฤ ออกเสียงเป็น รึ ถ้าประสมกับตัวอื่น เช่น คฤหาสน์ พฤศจิกายน มฤตยู หฤทัย
7. อักษร “ห” นำ ให้ตัด ห ทิ้งถ้าอักษร “ห” นำตัวอักษร ร ล ว ง ญ น ม เพราะไม่ออกเสียงพยัญชนะ “ห” แต่ออกเสียงพยัญชนะต้นตามตัวอักษรที่ตามหลัง “ห” [17] เช่น หรู ไหล หวี เหงา หญิง หนา หมู

3.3 การเข้ารหัสคำ

การเข้ารหัสคำคือการแปลงแต่ละตัวอักษรในคำไปเป็นรหัสเสียงโดยการเทียบเสียงจากตารางรหัสเสียง แล้วนำรหัสเสียงมาเรียงต่อกันเป็นรหัสคำ ประเด็นหนึ่งที่ต้องพิจารณาในหารหัสเสียงก็คือความสัมพันธ์ระหว่างตัวอักษรและรหัสเสียง สำหรับความสัมพันธ์แบบหนึ่งต่อหนึ่งซึ่งสามารถเทียบรหัสเสียงได้โดยตรงจากตาราง แต่ถ้าเป็นความสัมพันธ์แบบหนึ่งต่อหลายจะมีวิธีการพิจารณาดังนี้

1. <u>แบบหลายตัวอักษรต่อหนึ่งหน่วยเสียง</u> ในกรณีนี้จะมีตัวอักษรตัวเดียวที่ให้รหัสเสียง ส่วนตัวที่เหลือให้รหัสเป็น “_” (หมายถึงตัวอักษรที่ไม่ออกเสียง) ตัวอย่างเช่น คำภาษาอังกฤษ “king” จะมีรหัสเป็น kig_ โดยรหัสเสียง g เกิดจากกลุ่มตัวอักษร “ng” หรือเช่นคำไทย “เกรียง” จะมีรหัสเป็น lkr__g โดยรหัสเสียง l เกิดจากกลุ่มตัวอักษร “เ–ีย”

2. <u>แบบหลายหน่วยเสียงต่อหนึ่งตัวอักษร</u> ในบางกรณีสำหรับคำอังกฤษ ตัวอักษรตัวหนึ่งในคำอาจให้เสียงมากกว่า 1 เสียง เช่น ตัว “x” ใน “toxin” ให้ทั้งเสียง /k/ และ /s/ ในกรณีนี้

จะเลือกเพียงเสียงเดียวโดยให้มีรหัสเป็น tosin สำหรับคำไทยนั้นมีการใช้สระลดรูป เช่น ลดา (สระ –ะ ลดรูป) นก (สระ โ-ะ ลดรูป) สุนทร (สระ –อ ลดรูป) ในรหัสคำจะแทรกรหัสเสียงตามเสียงสระที่ลดรูปไป ซึ่งทำให้บางกรณีให้รหัสเสียงมากกว่า 1 รหัส ดังนั้นจากตัวอย่าง “ลดา” จะได้รหัสเป็น lada “นก” จะได้รหัสเป็น nok และ “สุนทร” จะได้รหัสเป็น sunton

นอกจากนี้ในงานวิจัยได้แบ่งการเข้ารหัสคำตามภาษาและการทับศัพท์ ได้แก่ คำไทย (เช่น เจริญ) คำอังกฤษทับศัพท์คำไทย (เช่น charoen) คำอังกฤษ (เช่น interface) และคำไทยทับศัพท์คำอังกฤษ (เช่น อินเตอร์เฟส) ดังนั้นจึงต้องมีตัวเข้ารหัสคำทั้งหมด 4 ชุดด้วยกัน

ในบางครั้งนั้นเราจะทราบว่าตัวอักษรที่กำลังพิจาณานั้นให้เสียงอย่างไร จะต้องพิจารณาตัวอักษรที่อยู่ข้างเคียงด้วย ดังนั้นเราจึงสามารถพิจารณาได้ว่าปัญหานี้เป็นปัญหาการจำแนกประเภท (Classification) ได้ ซึ่งในงานวิจัยนี้เสนอการใช้นิวรอลเน็ตเวิร์กและแบบจำลองฮิดเด็นมาร์คอฟในการเข้ารหัสคำ จำนวนอักษรที่ใช้พิจารณานั้นขึ้นกับว่าคำที่ต้องการเข้ารหัสนั้นเขียนอยู่ในรูปใด ถ้าเขียนในรูปอักษรไทย (ได้แก่ คำไทยและคำไทยทับศัพท์คำอังกฤษ) จะใช้จำนวนตัวอักษร 9 ตัว แต่ถ้าเขียนในรูปตัวอักษรอังกฤษ (ได้แก่ คำอังกฤษและคำอังกฤษทับศัพท์คำไทย) จะใช้ 7 ตัว ซึ่งประกอบด้วย ตัวอักษรที่กำลังพิจารณา 1 ตัว ที่เหลือเป็นตัวอักษรที่อยู่ข้างหน้าและตัวอักษรที่อยู่ข้างหลัง อย่างละเท่าๆกัน รูปที่ 3.1 และ รูปที่ 3.2 แสดงตัวอย่างการเข้ารหัสคำ “สุรเกียรติ” และ “surakiat” ด้านซ้ายมือคือลำดับของตัวอักษรประกอบด้วยตัวอักษรข้างเคียงและตัวอักษรที่กำลังพิจารณา โดยเครื่องหมาย ‘_’ หมายถึงอักษรว่าง (Blank) ส่วนด้านขวาคือรหัสเสียงของแต่ละลำดับโดยเครื่องหมาย ‘_’ หมายความว่าไม่มีการออกเสียงสำหรับลำดับตัวอักษรนั้น เมื่อนำรหัสเสียงจากทุกลำดับมาต่อกัน จากรูปที่ 3.1 จะได้รหัสคำเป็น “suralk___t_” และจากรูปที่ 3.2 จะได้รหัสคำเป็น “surakl_t” หลังจากนั้นนำรหัสคำที่ได้มาผ่านการประมวลผลภายหลังเช่นเดียวกับในงานวิจัย [12] และ [13] โดยตัดรหัส ‘_’ ออก แล้วย้ายรหัสที่แทนเสียงสระทั้งหมดไปอยู่ต่อท้ายรหัสที่แทนเสียงพยัญชนะ ดังนั้นสุดท้ายแล้วรหัสคำของตัวอย่างจะได้เป็น “srktual”

3.3.1 การเข้ารหัสคำด้วยนิวรอลเน็ตเวิร์ก

นิวรอลเน็ตเวิร์กที่ใช้เป็นแบบแบ็กพรอพาเกชัน (Backpropagation Neural Network) ซึ่งมี 3 ชั้น ดังแสดงในรูปที่ 3.3 โดยให้ข้อมูลขาเข้าจะเป็นตัวอักษรภาษาไทย ประกอบด้วยตัวอักษรที่ต้องการทราบรหัสเสียง และตัวอักษรที่อยู่ข้างเคียงกับมัน ข้างละ 4 ตัว รวมเป็น 9 ตัว (เนื่องจาก

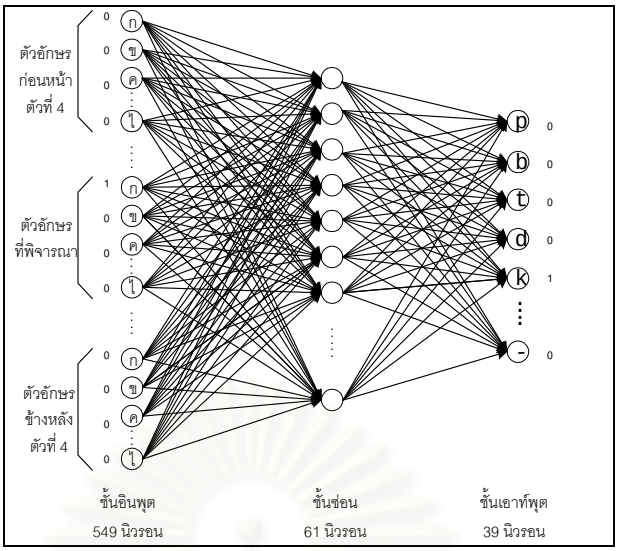
ข้อมูลเข้าอยู่ในรูปอักษรไทย จึงใช้จำนวนตัวอักษร 9 ตัว) ส่วนข้อมูลขาออกจะเป็นรหัสเสียงของข้อมูลขาเข้า (ตามที่ได้แสดงในตารางที่ 3.1 และ ตารางที่ 3.2)

ลำดับตัวอักษร	รหัส
_ , _ , _ , _ , ส , ุ , ร , เ , ก	s
_ , _ , _ , ส , ุ , ร , เ , ก , ี	u
_ , _ , ส , ุ , ร , เ , ก , ี , ย	r
_ , ส , ุ , ร , เ , ก , ี , ย , ร	a l
ส , ุ , ร , เ , ก , ี , ย , ร , ต	k
ุ , ร , เ , ก , ี , ย , ร , ต , ิ	_
ร , เ , ก , ี , ย , ร , ต , ิ , _	_
เ , ก , ี , ย , ร , ต , ิ , _ , _	_
ก , ี , ย , ร , ต , ิ , _ , _ , _	t
ี , ย , ร , ต , ิ , _ , _ , _ , _	_

รูปที่ 3.1 การเข้ารหัสคำโดยอาศัยการพิจารณาอักษรข้างเคียงสำหรับคำ “สุรเกียรติ”

ลำดับตัวอักษร	รหัส
_ , _ , _ , s , u , r , a	s
_ , _ , s , u , r , a , k	u
_ , s , u , r , a , k , i	r
s , u , r , a , k , i , a	a
u , r , a , k , i , a , t	k
r , a , k , i , a , t , _	l
a , k , i , a , t , _ , _	_
k , i , a , t , _ , _ , _	t

รูปที่ 3.2 การเข้ารหัสคำโดยอาศัยการพิจารณาอักษรข้างเคียงสำหรับคำ “surakiat”



รูปที่ 3.3 การเข้ารหัสคำด้วยแบ็กพรอพาเกชันนิวรอลเน็ตเวิร์ก

รายละเอียดโครงสร้างของนิวรอลเน็ตเวิร์คแต่ละชั้นมีดังต่อไปนี้

ชั้นอินพุต (Input Layer) อินพุตของนิวรอลเน็ตเวิร์กจึงได้มาจากการแทนแต่ละตัวอักษรที่พิจารณาด้วยจำนวนนิวรอนเท่ากับจำนวนอักขระในแต่ละภาษา โดยแต่ละนิวรอนแทนการปรากฏของตัวอักษรแต่ละตัว เพราะฉะนั้นชั้นอินพุตจึงมีจำนวนนิวรอนเท่ากับผลคูณระหว่างจำนวนอักขระของภาษาคูณและจำนวนอักษรที่ใช้พิจารณา จำนวนอักขระสำหรับกรณีคำที่สะกดด้วยตัวอักษรอังกฤษมีจำนวน 26 ตัว (a-z) จึงใช้จำนวนนิวรอนเป็น 26×7=182 นิวรอน และในกรณีคำที่สะกดด้วยตัวอักษรไทยมีจำนวน 61 ตัว (พยัญชนะและสระเดี่ยว) จึงใช้จำนวนนิวรอนเป็น 61×9=549 นิวรอน อินพุตของแต่ละนิวรอนมีค่าเป็น 0 หรือ 1

ชั้นซ่อน (Hidden Layer) จำนวนนิวรอนในชั้นนี้ต้องได้มาจากการทดลองซึ่งให้ค่าความถูกต้องสูงสุด โดยการทดลองนี้จะได้กล่าวถึงในภายหลัง

ชั้นเอาท์พุต (Output Layer) มีจำนวนนิวรอน เท่ากับจำนวนรหัสเสียงที่กำหนด โดยมี 33 รหัสเสียงสำหรับคำอังกฤษและคำไทยทับศัพท์คำอังกฤษ (ดูตารางที่ 3.1) ส่วนคำไทยและคำอังกฤษทับศัพท์คำไทยมี 35 รหัสเสียง (ดูตารางที่ 3.2) นิวรอนที่ให้ค่าเอาท์พุตสูงสุดจะเป็นรหัสเสียงที่เป็นคำตอบ ในกรณีคำไทยนั้นมีการใช้สระลดรูป ดังนั้นในขั้นการฝึกจะกำหนดให้มีค่าเป้าหมาย 2 ตัวที่มีค่าเป็น 1 ซึ่งแทนการมีคำตอบ 2 คำตอบ ส่วนในขั้นตอนการเข้ารหัสคำนั้นถ้านิวรอนที่ให้ค่าเอาท์พุตสูงสุด 2 อันดับแรกมีค่าต่างกันไม่เกิน 0.3 (ซึ่งกำหนดตามที่ระบุในงานวิจัย [9]) จะถือว่าทั้งสองนิวรอนนั้นเป็นคำตอบ

3.3.2 การเข้ารหัสคำด้วยแบบจำลองฮิดเด็นมาร์คอฟ

ในงานวิจัยนี้ใช้แบบจำลองฮิดเด็นมาร์คอฟแบบซ้ายไปขวา กำหนดให้แบบจำลองฮิดเด็นมาร์คอฟแต่ละแบบจำลองแทนการให้เสียงแต่ละเสียง ดังนั้นจำนวนแบบจำลองที่ใช้จึงเท่ากับจำนวนรหัสเสียงที่กำหนดไว้ นั่นคือ 33 แบบจำลองสำหรับคำอังกฤษและคำไทยทับศัพท์คำอังกฤษ และ 35 แบบจำลองสำหรับคำไทยและคำอังกฤษทับศัพท์คำไทย ข้อมูลที่เป็นค่าสังเกตคือลำดับ (Sequence) ของตัวเลขซึ่งเป็นหมายเลขของตัวอักษร แต่ละลำดับมีขนาดเป็น 9 สำหรับคำที่เขียนในรูปอักษรไทย และมีขนาดเป็น 7 สำหรับคำที่เขียนในรูปอักษรอังกฤษ ขนาดของลำดับนี้แทนความหมายของจำนวนตัวอักษรที่ใช้พิจารณา แต่ละลำดับจะถูกนำไปฝึกแบบจำลองที่สอดคล้องกับมัน เช่น ลำดับที่ให้รหัสเสียงเป็น ‘s’ ก็จะใช้ฝึกแบบจำลองสำหรับเสียง ‘s’ สำหรับหมายเลขของตัวอักษรนั้น ตัวอักษรภาษาอังกฤษใช้หมายเลข 1 ถึง 27 ตัวอักษรภาษาไทยใช้หมายเลข 1 ถึง 62 โดยหมายเลขสุดท้ายของแต่ละกรณี (27 และ62) ใช้แทนตัวอักษรว่าง รูปที่ 3.4 แสดงตัวอย่างลำดับหมายเลขตัวอักษรสำหรับคำ “สุรเกียรติ” ซึ่งจะสังเกตได้ว่าเนื่องจากคำนี้เป็นคำที่เขียนในรูปอักษรไทยจึงใช้ขนาดของลำดับเป็น 9 ส่วนรูปที่ 3.5 แสดงลำดับหมายเลขของคำ “surakiat” ซึ่งใช้ขนาดของลำดับเป็น 7 เนื่องจากเขียนในรูปตัวอักษรอังกฤษ

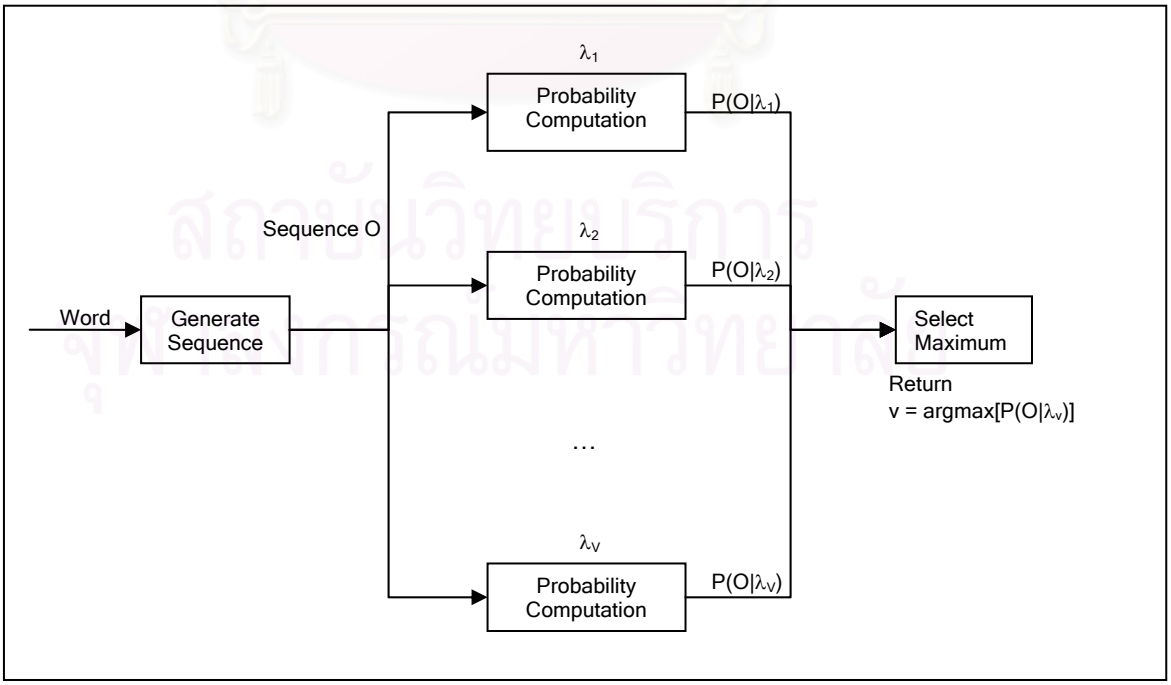
ลำดับตัวอักษร	ลำดับหมายเลขตัวอักษร
_ , _ , _ , _ , ส , ุ , ร , เ , ก	62 , 62 , 62 , 62 , 42 , 55 , 35 , 57 , 1
_ , _ , _ , ส , ุ , ร , เ , ก , ี	62 , 62 , 62 , 42 , 55 , 35 , 57, 1 , 52
_ , _ , ส , ุ , ร , เ , ก , ี , ย	62 , 62 , 42 , 55 , 35 , 57 , 1 , 52 , 34
_ , ส , ุ , ร , เ , ก , ี , ย , ร	62 , 42 , 55 , 35 , 57 , 1 , 52 , 34 , 35
ส , ุ , ร , เ , ก , ี , ย , ร , ต	42 , 55 , 35 , 57 , 1 , 52 , 34 , 35 , 21
ุ , ร , เ , ก , ี , ย , ร , ต , ิ	55 , 35 , 57 , 1 , 52 , 34 , 35 , 21 , 51
ร , เ , ก , ี , ย , ร , ต , ิ , _	35 , 57 , 1 , 52 , 34 , 35 , 21 , 51 , 62
เ , ก , ี , ย , ร , ต , ิ , _ , _	57 , 1 , 52 , 34 , 35 , 21 , 51 , 62 , 62
ก , ี , ย , ร , ต , ิ , _ , _ , _	1 , 52 , 34 , 35 , 21 , 51 , 62 , 62 , 62
ี , ย , ร , ต , ิ , _ , _ , _ , _	52 , 34 , 35 , 21 , 51 , 62 , 62 , 62 , 62

รูปที่ 3.4 ตัวอย่างการกำหนดลำดับของหมายเลขตัวอักษรสำหรับคำ “สุรเกียรติ”

ลำดับตัวอักษร	ลำดับหมายเลขตัวอักษร
_ , _ , _ , s , u , r , a	27 , 27 , 27 , 19 , 21 , 18 , 1
_ , _ , s , u , r , a , k	27 , 27 , 19 , 21 , 18 , 1 , 11
_ , s , u , r , a , k , i	27 , 19 , 21 , 18 , 1 , 11 , 9
s , u , r , a , k , i , a	19 , 21 , 18 , 1 , 11 , 9 , 1
u , r , a , k , i , a , t	21 , 18 , 1 , 11 , 9 , 1 , 21
r , a , k , i , a , t , _	18 , 1 , 11 , 9 , 1 , 21 , 27
a , k , i , a , t , _ , _	1 , 11 , 9 , 1 , 21 , 27 , 27
k , i , a , t , _ , _ , _	11 , 9 , 1 , 21 , 27 , 27 , 27

รูปที่ 3.5 ตัวอย่างการกำหนดลำดับของหมายเลขตัวอักษรสำหรับคำ “surakiat”

ในการสร้างรหัสคำจะนำแต่ละลำดับของตัวอักษร ไปทำการคำนวณค่าความน่าจะเป็นของลำดับนั้นด้วยขั้นตอนวิธีไปข้างหน้ากับทุกแบบจำลองรหัสเสียง ให้รหัสเสียงของลำดับนั้นคือรหัสเสียงแบบจำลองรหัสเสียงที่ให้ค่าความน่าจะเป็นมากที่สุด เมื่อนำรหัสเสียงที่ได้จากทุกลำดับมาเรียงต่อกันก็จะได้รหัสคำ ในรูปที่ 3.6 แสดงแผนภาพการเข้ารหัสคำ โดย O คือลำดับของตัวอักษรในคำ แต่ละ λ_i แทนแบบจำลองฮิดเด็นมาร์คอฟของรหัสเสียงแต่ละตัว รหัสเสียงที่เป็นคำตอบก็คือรหัส v ซึ่ง $P(O | \lambda_v)$ ให้ค่าสูงสุดจากทุก λ_i , $1 \leq i \leq V$



รูปที่ 3.6 การเข้ารหัสคำด้วยแบบจำลองฮิดเด็นมาร์คอฟ

ในงานวิจัยนี้ใช้ขั้นตอนวิธี Baum-welch ในการฝึกแบบจำลองฮิดเด็นมาร์คอฟ จำนวนสถานะและอันดับของแบบจำลองนั้นจะได้มาจากการทดลองซึ่งให้ค่าความถูกต้องสูงสุด การทดลองนี้จะได้กล่าวถึงในภายหลัง สำหรับการจัดการกับสระลดรูปที่ปรากฏในคำไทยนั้นในขั้นการฝึกจะให้ลำดับค่าสังเกตของกรณีสระลดรูปเป็นข้อมูลฝึกของทั้งสองแบบจำลอง เช่น ในรูปที่ 3.1 ลำดับค่าสังเกตที่สี่จะเป็นข้อมูลฝึกของแบบจำลองทั้งสำหรับรหัสเสียง ‘a’ และรหัสเสียง ‘I’ ส่วนในขั้นตอนการเข้ารหัสคำนั้นจะให้คำตอบเป็น 2 รหัสเสียงก็ต่อเมื่อ แบบจำลองที่ให้ค่าความน่าจะเป็นสูงสุด 2 อันดับแรกมีค่าต่างกันไม่เกิน log(0.3) (สาเหตุที่ใช้ฟังก์ชัน log เนื่องจากการคำนวณความน่าจะเป็นด้วยขั้นตอนวิธีไปข้างหน้านั้นอยู่ในรูปของลอการิทึม (Logarithm))

3.4 **สรุป**

ในบทนี้ได้กล่าวถึงวิธีการเข้ารหัสคำ ซึ่งประกอบด้วยการกำหนดรหัสคำซึ่งแทนเสียงของแต่ละตัวอักษรในภาษา การประมวลผลเบื้องต้นสำหรับคำภาษาไทยเพื่อลดความซับซ้อนในการเข้ารหัสคำ แนวคิดในการเข้ารหัสคำโดยพิจาณาว่าปัญหาการเข้ารหัสคำนั้นเป็นปัญหาการจำแนกประเภท และอธิบายถึงวิธีการการเข้ารหัสคำโดยใช้นิวรอลเน็ตเวิร์ก และแบบจำลองฮิดเด็นมาร์คอฟ

บทที่ 4

การค้นคืนข้ามภาษา

ขั้นตอนการค้นคืนข้ามภาษาประกอบด้วยการนำรหัสคำของคำที่เป็นคำถาม ไปเปรียบเทียบกับรหัสคำในดัชนีคำ ถ้าผลการเปรียบเทียบออกมาอยู่ในเกณฑ์ที่กำหนด ก็จะถือว่ารหัสคำทั้งสองมีเสียงอ่านตรงกัน และระบบจะคืนคำนั้นกลับไป ในบทนี้จะอธิบายวิธีการในการเปรียบเทียบรหัสคำ เกณฑ์ที่ใช้ในการเปรียบเทียบรหัสคำ การใช้ขั้นตอนวิธีเชิงพันธุกรรมในการเพิ่มประสิทธิภาพของการค้นคืน และการวัดผลการค้นคืน

4.1 การคำนวณความต่างของรหัสคำ

เนื่องจากรหัสคำที่ได้จากขั้นตอนการเข้ารหัสคำด้วยนิวรอลเน็ตเวิร์กนั้น แม้ว่าคู่คำจากทั้งสองภาษาจะออกเสียงเหมือนกัน ก็อาจมีรหัสไม่เหมือนกันทุกตัวอักษร ดังนั้นในการเปรียบเทียบรหัสคำ จึงต้องใช้วิธีการเปรียบเทียบแบบประมาณ (Approximate Matching) ด้วยเทคนิคระยะแก้ไขสั้นสุด (Minimum Edit Distance) [7] ซึ่งเป็นการคำนวณความคล้ายคลึงกันระหว่างสายอักขระ 2 สาย โดยคำนวณจากต้นทุน (Cost) ที่ใช้ในการเปลี่ยนสายอักขระหนึ่งไปเป็นอีกสายอักขระหนึ่ง ด้วยการเพิ่ม ลบ หรือแทนที่อักขระ

$$\text{Edit}\ (P_0, W_0) = 0$$

$$\text{Edit}\ (P_j, W_0) = \text{Edit}\ (P_{j-1}, W_0) + D(p_j)$$

$$\text{Edit}\ (P_0, W_k) = \text{Edit}\ (P_0, W_{k-1}) + D(w_k)$$

$$\text{Edit}\ (P_j, W_k) = \min \{ \text{Edit}\ (P_{j-1}, W_k) + D(p_j),$$
$$\text{Edit}\ (P_j, W_{k-1}) + D(w_k),$$
$$\text{Edit}\ (P_{j-1}, W_{k-1}) + R(p_j, w_k) \ \}$$

โดยที่

$P_j = p_1\ p_2\ p_3 \ldots\ p_j$ เป็นสายอักขระต้นแบบ มีความยาว j ตัวอักษร

$W_k = w_1\ w_2\ w_3 \ldots\ w_k$ เป็นสายอักขระเป้าหมาย มีความยาว k ตัวอักษร

$D(p_j)$ = ต้นทุนการเพิ่มหรือลบอักขระ p_j

$R(p_j, w_k)$ = ค่าจากตารางต้นทุนการแทนที่อักขระ p_j ด้วย w_k

รูปที่ 4.1 เทคนิคระยะแก้ไขสั้นที่สุด

ในงานวิจัยนี้ได้มีการปรับเปลี่ยนเทคนิคระยะแก้ไขสั้นสุดจากใน [5] โดยต้นทุนในการแทนที่อักขระจะได้มาจากตารางต้นทุนการแทนที่อักขระซึ่งเป็นตารางสำหรับกำหนดต้นทุนในการแทนที่อักขระคู่ใดๆ เทคนิคระยะแก้ไขสั้นที่สุดสามารถเขียนเป็นความสัมพันธ์เวียนบังเกิด (Recurrent Relation) ได้ตามรูปที่ 4.1

ตารางต้นทุนการแทนที่อักขระมีขนาด n × n โดย n คือจำนวนรหัสคำ ค่าต้นทุนในตารางสามารถกำหนดให้มีหลายระดับ ตารางนี้มีลักษณะสมมาตร (ค่าในแถวที่ i หลักที่ j เท่ากับค่าในแถวที่ j หลักที่ i) และค่าในแนวเส้นทแยงมุมเท่ากับศูนย์ ส่วนตารางต้นทุนการแทนที่อักขระมีขนาด 1 × n รูปที่ 4.2 แสดงตัวอย่างตารางต้นทุนการแทนที่และเพิ่ม/ลบอักขระ โดยตัวอักษร p, b, t, d ในตัวอย่าง หมายถึงรหัสเสียง

	p	b	t	d	…
p	0	1	3	3	…
b	1	0	4	2	…
t	3	4	0	2	…
d	3	2	2	0	…
…	…	…	…	…	…

(ก)

p	b	t	d	…
2	1	3	3	…

(ข)

รูปที่ 4.2 ตัวอย่างตารางต้นทุนการแทนที่อักขระและตารางต้นทุนการเพิ่ม/ลบอักขระ

4.2 เกณฑ์การเปรียบเทียบรหัสคำ

ค่าความแตกต่างระหว่างรหัสคำที่อยู่ในเกณฑ์ต่อไปนี้ จะถือว่าทั้งสองรหัสคำมีเสียงอ่านตรงกัน

$$\text{Edit}(P_m, W_n) \leq \alpha \times \text{Max}(m, n) \times \text{MaxEditCost} \qquad (4.1)$$

โดยที่ P_m คือรหัสคำความยาว m W_n คือรหัสคำความยาว n MaxEditCost คือค่าต้นทุนการแก้ไขอักขระที่มากที่สุด เช่น ถ้าต้นทุนการแทนที่มี 4 ระดับคือ 1, 2, 3, 4 และต้นทุนการเพิ่ม

หรือลบอักขระเป็น 2 แล้ว ค่า MaxEditCost จะเท่ากับ 4 ดังนั้น Max(m, n) × MaxEditCost จึงเท่ากับต้นทุนที่มากที่สุดที่ใช้ในการแก้ไขสายอักขระ นั่นคือเปลี่ยนทุกอักขระใน P ซึ่งมีความยาว m ไปเป็น W ซึ่งมีความยาว n ส่วน α คือระดับการยอมรับความแตกต่าง ซึ่งจะส่งผลต่อค่าแม่นยำและค่าเรียกคืนของระบบ [12] α เป็นจำนวนจริงมีค่าอยู่ในช่วง [0 , 1] โดยถ้ามีค่าเท่ากับ 0 จะหมายถึงว่ารหัสคำทั้งสองต้องเหมือนกันทุกตัว จึงจะยอมรับว่ารหัสคำทั้งสองมีเสียงอ่านตรงกัน (นั่นคือค่าความแตกต่างเท่ากับ 0) และถ้ามีค่าเท่ากับ 1 จะเป็นการยอมรับว่ารหัสคำทั้งสองมีเสียงอ่านตรงกันไม่ว่าค่าความแตกต่างของรหัสคำที่คำนวณออกจะมีค่ามากน้อยเพียงใด ตัวอย่างการคิดเกณฑ์การเปรียบเทียบรหัสคำมีดังนี้

ตัวอย่างที่ 4.1

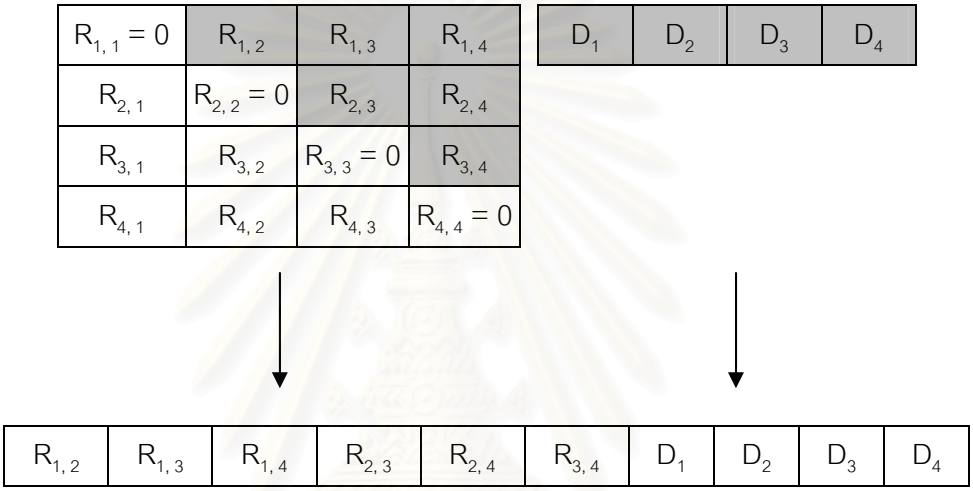
- กำหนดให้คำ “สวิส” มีรหัสคำเป็น swis ให้คำ “สวิตช์” มีรหัสคำเป็น swit
- ให้ R(s , t) = 1 MaxEditCost = 4 และ α = 0.15
- ดังนั้น Edit(swis , swit) =1 $\leq$ 0.15 × 4 × 4 จึงสรุปว่า “สวิส" และ “สวิตช์” มีเสียงอ่านตรงกัน

4.3 ขั้นตอนวิธีเชิงพันธุกรรมและการค้นคืน

เนื่องจากการสร้างตารางต้นทุนการแทนที่อักขระนี้มีความซับซ้อน และไม่มีหลักเกณฑ์ที่แน่นอนในการกำหนดค่าต้นทุนการแก้ไขอักขระว่าจะให้มีค่ามากหรือน้อยเพียงใด ทำให้การคำนวณความแตกต่างของรหัสคำอาจไม่ได้ค่าที่เหมาะสมและส่งผลให้การค้นคืนมีความถูกต้องลดลงไป จากตัวอย่างที่ 4.1 ถ้ากำหนดให้ R(s , t) = 3 แล้ว จะได้ว่า Edit(swis , swit) = 3 ซึ่งมากกว่าเกณฑ์ที่กำหนดไว้ ทำให้ตัดสินผิดพลาดไปว่าคำ “สวิส” และ “สวิตช์” มีเสียงอ่านไม่ตรงกันทั้งที่จริงแล้วตรงกัน ดังนั้นในงานวิจัยนี้จึงใช้ขั้นตอนวิธีเชิงพันธุกรรม เข้ามาช่วยในการสร้างตารางต้นทุนการแก้ไขอักขระ เนื่องจากขั้นตอนวิธีเชิงพันธุกรรมเป็นขั้นตอนวิธีที่สามารถใช้กับปัญหาเกี่ยวกับการค้นหาและการหาค่าเหมาะสุด (Search and optimization) ได้ [4]

ในการใช้ขั้นตอนวิธีเชิงพันธุกรรมสำหรับงานวิจัยนี้นั้น เราแทนตารางต้นทุนการแก้ไขอักขระ (ซึ่งเป็นคำตอบที่เราต้องการ) ในรูปของโครโมโซม และให้ฟังก์ชันจุดประสงค์ (Objective function) คือการวัดผลการค้นคืน F1 ซึ่งใช้ตารางต้นทุนแก้ไขอักขระจากโครโมโซมในการคำนวณระยะแก้ไขสั้นสุด คำตอบที่ดีที่สุดที่หาได้ก็จะเป็น ตารางต้นทุนการแก้ไขอักขระที่ให้ค่า F1 ที่มากที่สุด

โครโมโซมที่ใช้จะประกอบด้วย 2 ส่วนคือ ส่วนต้นทุนการแทนที่อักขระ และส่วนต้นทุนการเพิ่ม/ลบอักขระ รูปที่ 4.3 แสดงตัวอย่างการแปลงรูปแบบระหว่างตารางต้นทุนการแทนที่อักขระขนาด 4x4 และต้นทุนการเพิ่ม/ลบอักขระขนาด 1x4 (รูปบน) และโครโมโซมซึ่งมีความยาวเท่ากับ 10 (รูปล่าง) จากในรูป $R_{i,j}$ แทนต้นทุนการแทนที่อักขระ i ด้วยอักขระ j และ D_i คือต้นทุนการเพิ่มหรือลบอักขระ i ส่วนที่แรเงาหมายถึงส่วนที่นำไปใช้ในการสร้างโครโมโซม ซึ่งจะเห็นได้ว่าค่าต้นทุนการแทนที่อักขระใช้นำเฉพาะส่วน $R_{i,j}$ ซึ่ง $i < j$ เท่านั้น เนื่องจากตารางต้นทุนการแทนที่อักขระนี้มีความสมมาตรดังที่ได้กล่าวมาแล้ว



รูปที่ 4.3 ตัวอย่างรูปแบบของโครโมโซมจากตารางต้นทุนการแทนที่อักขระขนาด 4x4

ในงานวิจัยนี้ใช้การไขว้เปลี่ยนแบบยูนิฟอร์ม (Uniform crossover) ซึ่งค่าของแต่ละยีนของโครโมโซมลูกได้มาจากการสุ่มเลือกจากยีนที่ตำแหน่งเดียวกันของโครโมโซมพ่อแม่ และทำการกลายพันธุ์โดยแทนที่ค่าเดิมของยีนเป็นค่าที่ได้จากการสุ่มเลือกสมาชิกของเซ็ตของค่าต้นทุนการแทนที่อักขระ โดยค่าใหม่นี้ต้องไม่ซ้ำกับค่าเดิม

ฟังก์ชันจุดประสงค์ที่ใช้คือการวัดค่าผลการค้นคืน F1 ซึ่งทำได้โดยนำคำ w_i แต่ละคำไปทำการค้นคืนโดยเปรียบเทียบรหัสคำกับทุกคำ w_j ในชุดข้อมูล W การเปรียบเทียบรหัสคำนี้ใช้เทคนิคระยะแก้ไขสั้นสุดซึ่งใช้ตารางต้นทุนการแทนที่อักขระจากจีโนมสำหรับการหาค่าต้นทุนการแทนที่อักขระ เมื่อเปรียบเทียบ w_i กับทุก w_j แล้วให้นับจำนวนคำที่ผลการเปรียบเทียบออกมาว่ามีเสียงอ่านตรงกัน และจำนวนคำที่มีเสียงอ่านตรงกันตามที่เป็นจริง แล้วคำนวณค่าตัววัด F1 ผลลัพธ์ของฟังก์ชันคือค่าเฉลี่ยของตัววัด F1 จากทุก w_i โดยอธิบายได้ดังนี้

กำหนดให้

W คือ เซ็ตของคำทั้งหมดที่ใช้

T คือ ตารางต้นทุนการแทนที่อักขระ

Edit(v, w, T) คือ ระยะแก้ไขสั้นที่สุดระหว่างรหัสคำของ v และ w โดยใช้ตาราง T สำหรับต้นทุนการแทนที่อักขระ

Rl คือ จำนวนคำที่เกี่ยวข้อง (ซึ่งหมายถึงคำที่มีเสียงอ่านตรงกันจริง)

Rt คือ จำนวนคำที่คืนกลับมา (ซึ่งหมายถึงคำที่ผ่านเกณฑ์การเปรียบเทียบรหัสคำ โดยอาจมีเสียงอ่านตรงกันหรือไม่ก็ได้)

Rr คือ จำนวนคำที่เกี่ยวข้องที่คืนกลับมา

Rl(w) คือ จำนวนคำที่เกี่ยวข้องทั้งหมดของคำ w

isRl(v, w) = true ; v และ w เป็นคำที่เกี่ยวข้องกัน

= false ; v และ w ไม่ใช่คำที่เกี่ยวข้องกัน

Prec คือ ค่าแม่นยำ

Recall คือ ค่าเรียกคืน

F1 คือ ค่าตัววัด F1

วิธีการคำนวณของฟังก์ชันจุดประสงค์สามารถเขียนได้ดังรูปที่ 4.4

4.4 สรุป

ในบทนี้ได้กล่าวถึงการค้นคืนข้ามภาษา โดยได้อธิบายถึงการคำนวณความต่างของรหัสคำด้วยการเปรียบเทียบเชิงประมาณด้วยการดัดแปลงเทคนิคระยะแก้ไขสั้นสุดโดยให้ต้นทุนการแก้ไขอักขระมีหลายระดับซึ่งได้จากตารางต้นทุนการแก้ไขอักขระ จากนั้นได้อธิบายถึงเกณฑ์การเปรียบเทียบรหัสคำเพื่อตัดสินว่ารหัสคำสองรหัสที่เปรียบเทียบกันนั้นมีเสียงอ่านตรงกันหรือไม่ และสุดท้ายได้อธิบายถึงการใช้ขั้นตอนวิธีเชิงพันธุกรรมในการหาตารางต้นทุนการแก้ไขอักขระที่เหมาะสมเพื่อให้นำไปใช้ในการเปรียบเทียบรหัสคำได้แม่นยำมากขึ้น

```
function Objective(g : genome)
SumF1 = 0
Decode g into T
for each w_i ∈W
   Rr = 0
   Rt = 0
   Rl = Rl(w_i)
   for each w_j ∈W
      if(edit(w_i, w_j, t) is not greater than thresold)
         Rt = Rt + 1
         if(isRl(w_i, w_j))
            Rr = Rr + 1
         End if
      end if
      Prec = (Rr / Rt) * 100
      Recall = (Rr / Rl) * 100
      F1 = (2 * Prec * Recall) / (Prec + Recall)
   end for
   SumF1 = SumF1 + F1
end for
AvgF1 = (SumF1/ |W|)
return AvgF1
end function
```

รูปที่ 4.4 ฟังก์ชันจุดประสงค์เพื่อวัดผลการค้นคืนในขั้นตอนวิธีเชิงพันธุกรรม

บทที่ 5

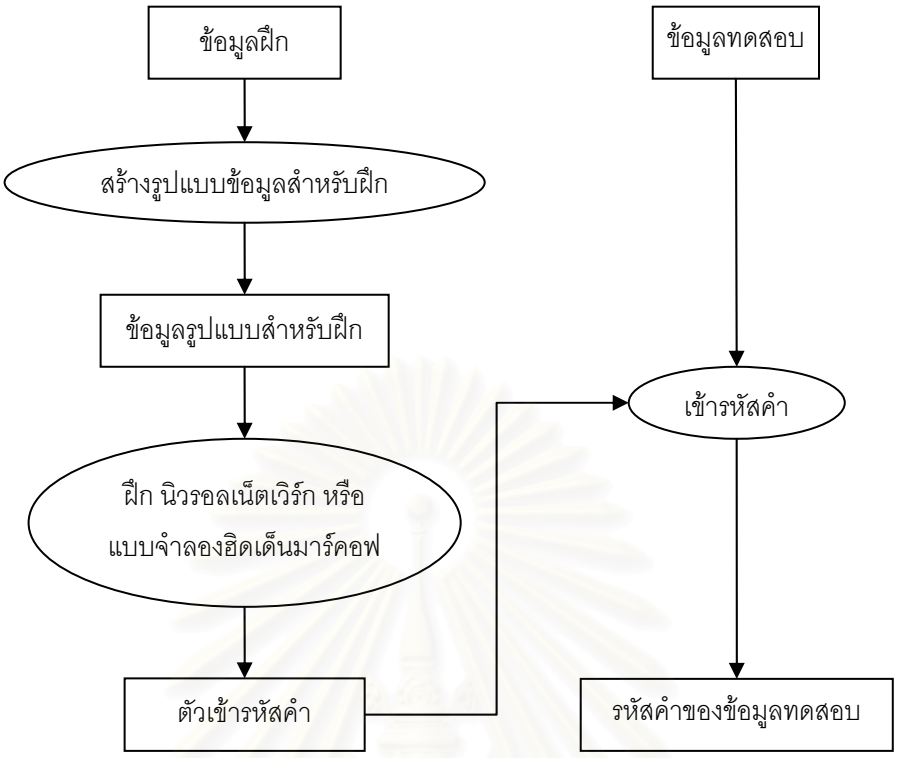
การทดลอง

5.1 วิธีการทดลอง

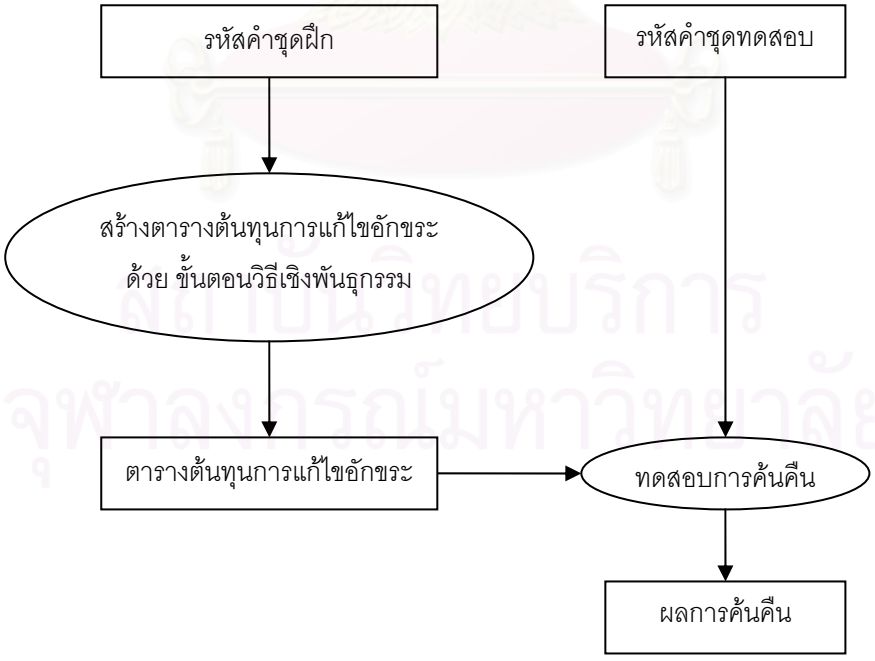
ในกรณีคำไทยทับศัพท์คำอังกฤษ ใช้ชุดของคำอังกฤษและคำทับศัพท์ที่ตรงกัน ซึ่งเป็นคำนามเฉพาะ คำศัพท์วิทยาศาสตร์ คำศัพท์คณิตศาสตร์ และคำศัพท์เคมี จำนวน 1,876 คู่ และกรณีคำอังกฤษทับศัพท์คำไทยใช้ชื่อและชื่อสกุลทั้งภาษาไทยและภาษาอังกฤษที่ตรงกันของนิสิต จำนวน 2,000 คู่ ทำการกำหนดรหัสคำของแต่ละคำศัพท์ไว้เพื่อใช้ในการฝึกสอนและเปรียบเทียบความถูกต้องเมื่อทำการทดลอง โดยยึดหลักเกณฑ์ทางภาษาศาสตร์ใน [15] และ [16] ข้อมูลถูกแบ่งเป็นชุดข้อมูลฝึก (Training set) และชุดข้อมูลทดสอบ (Test set) ชุดข้อมูลฝึกจะถูกนำไปสอนนิวรอลเน็ตเวิร์ก หรือแบบจำลองฮิดเด็นมาร์คอฟเพื่อสร้างตัวรหัสคำ แล้วใช้ตัวเข้ารหัสคำที่ได้ มาเข้ารหัสคำกับข้อมูลทั้งหมด ดังแสดงในรูปที่ 5.1 จากนั้นนำรหัสคำของชุดฝึกไปใช้ในการหาตารางต้นทุนการแก้ไขอักขระด้วยขั้นตอนวิธีเชิงพันธุกรรม สุดท้ายแล้วนำรหัสคำของชุดทดสอบและตารางต้นทุนที่ได้ ไปทดสอบการค้นคืนและบันทึกผลที่ได้ ดังแสดงในรูปที่ 5.2

เพื่อให้การทดลองไม่โน้มเอียงกับการแบ่งชุดฝึกกับชุดทดสอบ จึงใช้วิธี K–fold cross validation [2] ซึ่งทำโดยแบ่งข้อมูลทั้งหมดออกเป็น K ส่วนเท่าๆกัน ทำการทดลองทั้งหมด K ครั้ง ในแต่ละครั้งจะเลือกหนึ่งส่วนเป็นชุดทดสอบ ส่วนที่เหลือ K–1 ส่วนจะถูกใช้เป็นชุดฝึก จากนั้นนำผลการทดลองที่ได้จากการทดลองทั้งหมด K ครั้งมาหาค่าเฉลี่ย ในการทดลองนี้ได้แบ่งข้อมูลออกเป็นส่วนๆ ให้แต่ละส่วนมีคำประมาณ 400 คู่

การทดลองในงานวิจัยนี้ประกอบด้วย การทดลองเพื่อหาจำนวนนิวรอนในชั้นซ่อนที่เหมาะสมสำหรับนิวรอลเน็ตเวิร์ก การทดลองเพื่อหาจำนวนสถานะและอันดับที่เหมาะสมสำหรับแบบจำลองฮิดเด็นมาร์คอฟ และการทดลองขั้นตอนวิธีเชิงพันธุกรรมเพื่อหาตารางต้นทุนการแก้ไขอักขระที่เหมาะสมสำหรับการค้นคืน ในหัวข้อต่อไปนี้จะกล่าวถึงแต่ละการทดลองและผลการทดลองที่ได้



รูปที่ 5.1 วิธีการทดลองในขั้นตอนการสร้างตัวเข้ารหัสคำ



รูปที่ 5.2 วิธีการทดลองในขั้นตอนการใช้ขั้นตอนการสร้างตารางต้นทุนการแก้ไขอักขระ

5.2 การเข้ารหัสคำด้วยนิวรอลเน็ตเวิร์ก

ในการใช้งานนิวรอลเน็ตเวิร์กนั้นจำนวนนิวรอนในชั้นซ่อนเป็นปัจจัยหนึ่งที่มีผลต่อความถูกต้องของการรู้จำ ดังนั้นจึงได้ทำการทดลองเพื่อหาจำนวนนิวรอนที่เหมาะสมโดยกำหนดจำนวนนิวรอนต่างๆกัน ได้แก่ 10 50 100 150 และ 200 ในการทดลองนี้กำหนดอัตราการเรียนรู้เป็น 0.005 ค่าโมเมนตัมเป็น 0.95 และใช้จำนวนรอบในการฝึกนิวรอลเน็ตเวิร์ก 300 รอบ ผลการทดลองในตารางที่ 5.1 และตารางที่ 5.2 แสดงให้เห็นว่าเมื่อใช้จำนวนนิวรอนในชั้นซ่อนเป็น 200 สำหรับคำไทย และ 10 สำหรับคำอังกฤษทับศัพท์คำไทย จะให้ค่าความถูกต้องสูงสุดคือ 85.00% และ 91.41% ตามลำดับ และจากผลการทดลองในตารางที่ 5.2 จำนวนนิวรอนที่ให้ความถูกต้องสูงสุดสำหรับทั้งคำอังกฤษและสำหรับคำไทยทับศัพท์คำอังกฤษคือ 150 โดยให้ความถูกต้องเป็น 75.00% และ 91.31% ตามลำดับ ทั้งนี้ค่าความถูกต้องในตารางเป็นค่าเฉลี่ยที่ได้จากการทดลองจากข้อมูลทุกชุด

ตารางที่ 5.1 ค่าความถูกต้องเมื่อใช้จำนวนนิวรอนในชั้นซ่อนต่างๆกัน
สำหรับข้อมูลคำไทยและคำอังกฤษทับศัพท์คำไทย

จำนวนนิวรอนในชั้นซ่อน	ความถูกต้อง (เปอร์เซ็นต์)	
	คำไทย	คำอังกฤษทับศัพท์คำไทย
10	81.65	91.41
50	83.93	91.22
100	84.70	91.18
150	84.98	91.20
200	85.00	91.22

5.3 การเข้ารหัสคำด้วยแบบจำลองฮิดเด็นมาร์คอฟ

การทดลองนี้เป็นการทดลองเพื่อหาจำนวนสถานะและอันดับที่เหมาะสมสำหรับแบบจำลองฮิดเด็นมาร์คอฟ ผู้วิจัยได้ทดลองใช้จำนวนสถานะและอันดับต่างๆกันดังตารางที่ 5.3 และตารางที่ 5.4 จากตารางที่ 5.3 ซึ่งแสดงผลการทดลองสำหรับคำไทยและคำอังกฤษทับศัพท์คำไทย ค่าความถูกต้องสูงสุดสำหรับคำไทยคือ 74.70% เมื่อใช้จำนวนสถานะเป็น 30 และอันดับเป็น 5 ส่วนกรณีคำอังกฤษทับศัพท์คำไทยได้ค่าความถูกต้องสูงสุดคือ 91.57% เมื่อใช้จำนวนสถานะเป็น 30 และอันดับเป็น 10 จากตารางที่ 5.4 เมื่อใช้จำนวนสถานะเป็น 40 และอันดับเป็น 5

สำหรับคำไทยทับศัพท์คำอังกฤษ ได้ค่าความถูกต้องสูงสุดคือ 87.13% และเมื่อใช้จำนวนสถานะเป็น 30 และอันดับเป็น 5 สำหรับคำอังกฤษ ได้ค่าความถูกต้องสูงสุดคือ 79.14%

ตารางที่ 5.2 ค่าความถูกต้องเมื่อใช้จำนวนนิวรอนในชั้นซ่อนต่างๆกัน
สำหรับข้อมูลคำอังกฤษและคำไทยทับศัพท์คำอังกฤษ

จำนวนนิวรอนในชั้นซ่อน	ความถูกต้อง (เปอร์เซ็นต์)	
	คำอังกฤษ	คำไทยทับศัพท์คำอังกฤษ
10	71.80	88.51
50	73.70	91.11
100	74.12	91.00
150	75.00	91.31
200	74.94	91.23

ตารางที่ 5.3 ค่าความถูกต้องเมื่อใช้จำนวนสถานะและอันดับต่างๆ
สำหรับข้อมูลคำไทยและคำอังกฤษทับศัพท์คำไทย

จำนวนสถานะ	อันดับ	ความถูกต้อง (เปอร์เซ็นต์)	
		คำไทย	คำอังกฤษทับศัพท์คำไทย
10	5	48.05	78.66
10	10	43.82	77.30
20	5	67.96	90.10
20	10	65.92	89.52
30	5	74.70	91.07
30	10	71.25	91.57
40	5	74.26	90.61
40	10	73.41	91.48
50	5	73.61	90.29
50	10	74.29	90.89

ตารางที่ 5.4 ค่าความถูกต้องเมื่อใช้จำนวนสถานะและอันดับต่างๆ
สำหรับข้อมูลคำอังกฤษและคำไทยทับศัพท์คำอังกฤษ

จำนวนสถานะ	อันดับ	ความถูกต้อง (เปอร์เซ็นต์)	
		คำอังกฤษ	คำไทยทับศัพท์คำอังกฤษ
10	5	68.76	59.05
10	10	67.07	50.69
20	5	77.21	84.20
20	10	78.61	77.01
30	5	79.14	85.61
30	10	78.67	84.12
40	5	77.00	87.13
40	10	79.02	86.18
50	5	77.29	86.64
50	10	78.06	85.70

5.4 วิเคราะห์ผลการทดลองการเข้ารหัสคำ

เมื่อนำผลการทดลองสำหรับคำอังกฤษมาวิเคราะห์ พบว่ามีกลุ่มของรหัสเสียงจำนวนหนึ่งซึ่งความผิดพลาดในการจำแนกนั้นสามารถสังเกตได้ชัดเจนเนื่องจากบางตัวอักษรสามารถออกเสียงได้หลายแบบ ได้แก่ ตัวอักษร “a” สามารถให้รหัสเสียงได้ทั้ง “a” “w” และ “e” เช่นในคำ bar bad และ blade ตัวอักษร “e” สามารถให้รหัสเสียงได้ทั้ง “e” และ “i” เช่นในคำ bed และ begin ตัวอักษร “i” สามารถให้รหัสเสียงได้ทั้ง “i” และ “!” เช่นในคำ bit และ bios ดังนั้นปัญหานี้จึงมีผลกระทบต่อการรู้จำของรหัสเสียง a e i w ! โดยได้ผลรู้จำถูกต้องเฉลี่ยประมาณ 69% 77% 82% 67% และ 70% ตามลำดับจากการใช้นิวรอลเน็ตเวิร์ก และประมาณ 61% 66% 70% 61% และ 68% ตามลำดับจากการใช้แบบจำลองฮิดเด็นมาร์คอฟ

ผลการทดลองสำหรับคำไทยนั้นมีข้อสังเกตจากการวิเคราะห์ดังนี้ ประการแรกคือสระลดรูป ซึ่งพบว่าสามารถจำแนกกรณีสระลดรูป (ซึ่งมีจำนวนโดยเฉลี่ยประมาณ 11% ของข้อมูลทั้งหมด) ได้ถูกต้องเพียงประมาณ 76% ด้วยนิวรอลเน็ตเวิร์ก และประมาณ 19% ด้วยฮิดเด็นมาร์คอฟ ประการที่สองคือการจำแนกพยัญชนะต้นและตัวสะกด ตัวอักษรไทยบางตัวนั้นสามารถมีรหัสเสียงได้สองแบบ เช่น “จ” มีรหัสเสียงเป็น “t” เมื่อเป็นตัวสะกด และเป็น “c” เมื่อเป็น

พยัญชนะต้น ทำให้บางครั้งตัวเข้ารหัสคำจำแนกรหัสเสียงผิดไป เช่นเข้ารหัสคำ “สัจจา” เป็น satca หรือ sacca (ที่ถูกต้องคือ satca) หรือบางครั้งหากค่าเอาท์พุตของทั้งสองรหัส (ค่าของเอาท์พุตนิวรอนในนิวรอลเน็ตเวิร์ก หรือค่าความน่าจะเป็นที่คำนวณจากแบบจำลองฮิดเด็นมาร์คอฟ) มีความใกล้เคียงกันมาก ก็จะให้ทั้งสองรหัสเป็นคำตอบทั้งๆที่ต้องเป็นคำตอบเดียวจึงจะถูกต้อง ตัวอย่างเช่น คำว่า “สัจจา” อาจได้รหัสคำเป็น sactca หรือ satcta ซึ่งกรณีแบบนี้เป็นผลข้างเคียงจากการใช้วิธีการจัดการกับสระลดรูปโดยยอมให้ตัวเข้ารหัสคำสามารถให้คำตอบได้สองคำตอบ จากการวิเคราะห์ความผิดพลาดเฉพาะกรณีที่ต้องมีเพียงหนึ่งคำตอบ พบว่าประมาณ 67% (จากนิวรอลเน็ตเวิร์ก) และ 61% (จากฮิดเด็นมาร์คอฟ) ของความผิดพลาดนั้นมาจากการถูกจำแนกให้มีสองคำตอบ

สำหรับสำหรับผลการทดลองของคำไทยทับศัพท์คำอังกฤษและคำอังกฤษทับศัพท์คำไทยนั้น เมื่อวิเคราะห์แล้วพบว่าไม่มีจุดสังเกตที่ชัดเจนเท่ากับการทดลองกับข้อมูลทั้งสองชุดที่ได้กล่าวไว้แล้ว

5.5 ขั้นตอนวิธีเชิงพันธุกรรมและการค้นคืน

การทดลองนี้เป็นการนำขั้นตอนวิธีเชิงพันธุกรรมในการหาตารางต้นทุนการแก้ไขอักขระเพื่อการค้นคืนข้ามภาษา โดยกำหนดต้นทุนการแก้ไขอักขระตามตารางที่ 5.5 จากตาราง ในแบบที่ 1 นั้นต้นทุนการแก้ไขอักขระมีระดับเดียว ส่วนแบบที่ 2 3 และ 4 ใช้ต้นทุนหลายระดับและขั้นตอนวิธีเชิงพันธุกรรมในการหาตารางต้นทุนการแก้ไขอักขระ ข้อมูลที่ใช้ทดลองเป็นรหัสคำที่ได้มาจากการใช้นิวรอลเน็ตเวิร์กและแบบจำลองฮิดเด็นมาร์คอฟ การทดลองนี้แบ่งออกเป็นสองส่วน ส่วนแรกเป็นการทดลองสำหรับกรณีคำไทยทับศัพท์คำอังกฤษซึ่งใช้ข้อมูลรหัสคำของคำอังกฤษและคำไทยทับศัพท์คำอังกฤษในการค้นคืนข้ามภาษา ส่วนที่สองเป็นการทดลองสำหรับกรณีอังกฤษทับศัพท์คำไทยซึ่งใช้ข้อมูลรหัสคำของคำไทยและคำอังกฤษทับศัพท์คำไทย ในทุกการทดลองกำหนดค่าระดับการยอมรับความแตกต่างของรหัสคำ (ค่า α จากสมการ 4.1) เป็น 0.13 ขนาดของประชากร (Population size) เป็น 30 ความน่าจะเป็นของการไขว้เปลี่ยน (Crossover probability) มีค่า 0.80 ความน่าจะเป็นของการกลายพันธุ์ (Mutation probability) มีค่า 0.005 และจำนวนรุ่น (Number of generation) เป็น 1000

ตารางที่ 5.5 การกำหนดต้นทุนการแก้ไขอักขระในการทดลอง

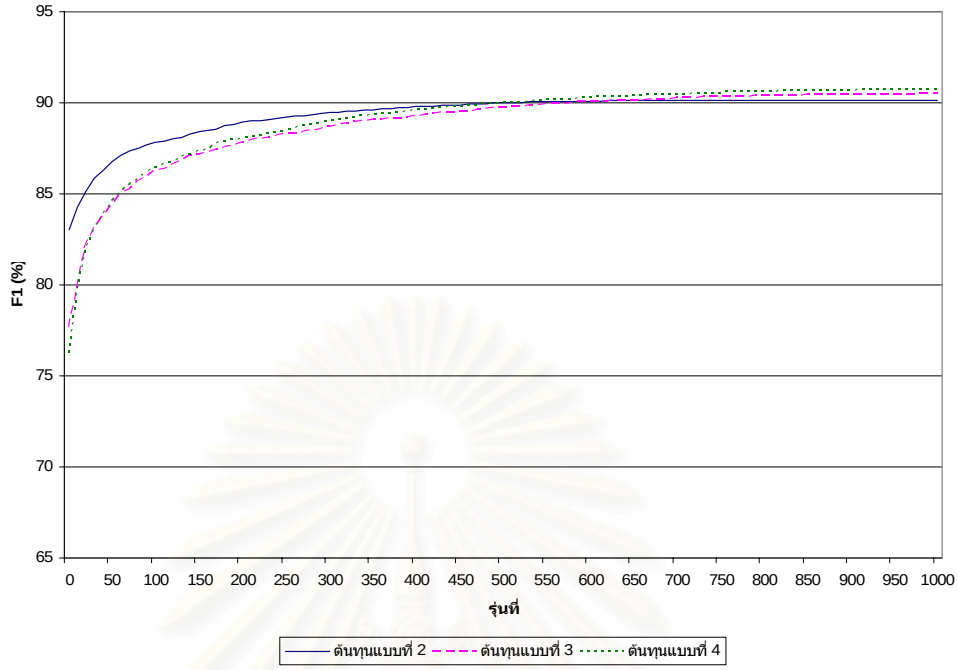
แบบที่	ค่าต้นทุน
1	1
2	1, 2, 3
3	1, 2, 3, 4
4	1, 2, 3, 4, 5

5.5.1 ผลการทดลองกรณีคำไทยทับศัพท์คำอังกฤษ

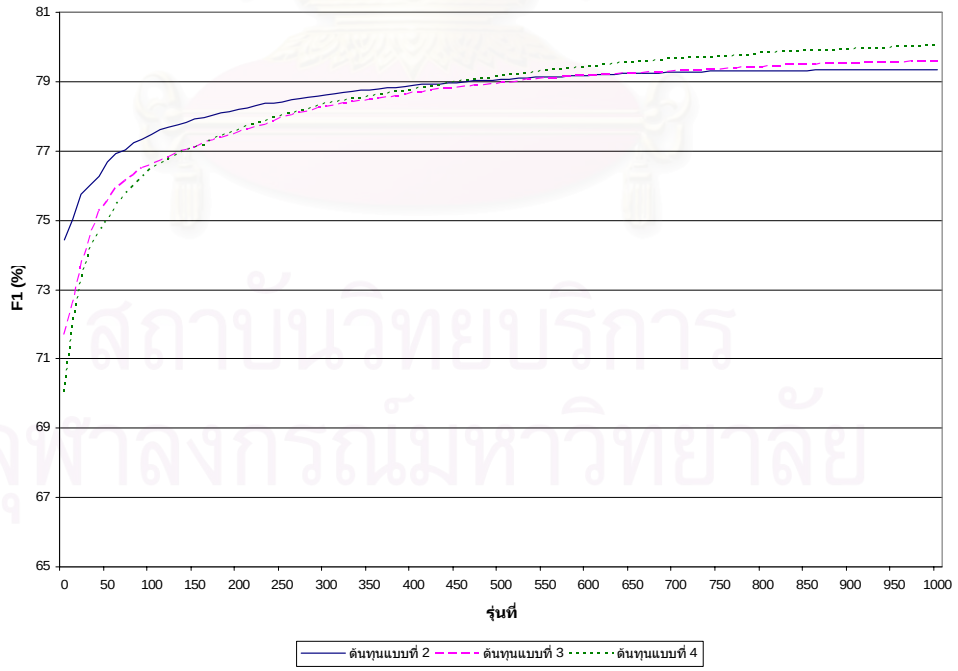
ตารางที่ 5.6 แสดงผลการทดลองกรณีคำไทยทับศัพท์คำอังกฤษโดยให้ต้นทุนการแก้ไขอักขระมีค่าต่างๆตามที่ได้ระบุในตารางที่ 5.5 เมื่อใช้ตัวเข้ารหัสคำเป็นนิวรอลเน็ตเวิร์กและแบบจำลองฮิดเด็นมาร์คอฟ ซึ่งจะเห็นได้ว่าการกำหนดให้ต้นทุนการแก้ไขอักขระมีค่าหลายระดับและใช้ขั้นตอนวิธีเชิงพันธุกรรมในการหาค่าเหมาะสุดของต้นทุน ให้ผลการค้นคืนที่ดีกว่าการกำหนดให้ต้นทุนการแก้ไขอักขระมีค่าระดับเดียว โดยกรณีที่ใช้นิวรอลเน็ตเวิร์กร่วมกับขั้นตอนวิธีเชิงพันธุกรรม ได้ผลการค้นคืน F1 เป็น 0.9107 0.9114 และ 0.9160 ส่วนกรณีที่ใช้แบบจำลองฮิดเด็นมาร์คอฟร่วมกับขั้นตอนวิธีเชิงพันธุกรรม ได้ผลการค้นคืน F1 เป็น 0.7902 0.7881 และ 0.7926 รูปที่ 5.3 และรูปที่ 5.4 แสดงผลการค้นคืนเฉลี่ยเมื่อใช้โครโมโซมที่ดีที่สุดในแต่ละรุ่นของขั้นตอนวิธีเชิงพันธุกรรมสำหรับการเข้ารหัสคำด้วยนิวรอลเน็ตเวิร์กและแบบจำลองฮิดเด็นมาร์คอฟตามลำดับ

ตารางที่ 5.6 ผลการทดลองกรณีคำไทยทับศัพท์คำอังกฤษ เมื่อให้ค่าต้นทุนเป็นแบบต่างๆ และใช้การเข้ารหัสคำด้วยนิวรอลเน็ตเวิร์กและแบบจำลองฮิดเด็นมาร์คอฟ

ต้นทุนแบบที่	นิวรอลเน็ตเวิร์ก			แบบจำลองฮิดเด็นมาร์คอฟ		
	ค่าแม่นยำ	ค่าเรียกคืน	F1	ค่าแม่นยำ	ค่าเรียกคืน	F1
1	0.9869	0.7505	0.8526	0.9897	0.5936	0.7419
2	0.9637	0.8631	0.9106	0.9552	0.6740	0.7902
3	0.9625	0.8655	0.9114	0.9518	0.6727	0.7881
4	0.9635	0.8730	0.9160	0.9583	0.6759	0.7926



รูปที่ 5.3 ผลการค้นคืนคำไทยทับศัพท์คำอังกฤษเมื่อใช้โครโมโซมที่ดีที่สุดในแต่ละรุ่นของขั้นตอนวิธีเชิงพันธุกรรมสำหรับการเข้ารหัสคำด้วยนิวรอลเน็ตเวิร์ก



รูปที่ 5.4 ผลการค้นคืนคำไทยทับศัพท์คำอังกฤษเมื่อใช้โครโมโซมที่ดีที่สุดในแต่ละรุ่นของขั้นตอนวิธีเชิงพันธุกรรมสำหรับการเข้ารหัสคำด้วยแบบจำลองฮิดเด็นมาร์คอฟ

5.5.2 ผลการทดลองกรณีคำอังกฤษทับศัพท์คำไทย

ตารางที่ 5.7 แสดงผลการทดลองกรณีคำอังกฤษทับศัพท์คำไทย ในทำนองเดียวกับกรณีคำไทยทับศัพท์คำอังกฤษ การใช้ขั้นตอนวิธีเชิงพันธุกรรมในการหาค่าเหมาะสุดของต้นทุนการแก้ไขอักขระที่มีหลายระดับ ให้ผลการค้นคืนที่ดีกว่าการกำหนดให้ต้นทุนการแก้ไขอักขระมีค่าระดับเดียว โดยกรณีที่ใช้นิวรอลเน็ตเวิร์กและขั้นตอนวิธีเชิงพันธุกรรม ได้ผลการค้นคืน F1 เป็น 0.9576 0.9565 และ 0.9559 ส่วนกรณีที่ใช้แบบจำลองฮิดเด็นมาร์คอฟและขั้นตอนวิธีเชิงพันธุกรรม ได้ผลการค้นคืน F1 เป็น 0.8104 0.8107 และ 0.8119 รูปที่ 5.5 และรูปที่ 5.6 แสดงผลการค้นคืนเฉลี่ยเมื่อใช้โครโมโซมที่ดีที่สุดในแต่ละรุ่นของขั้นตอนวิธีเชิงพันธุกรรมสำหรับการเข้ารหัสคำด้วยนิวรอลเน็ตเวิร์กและแบบจำลองฮิดเด็นมาร์คอฟตามลำดับ

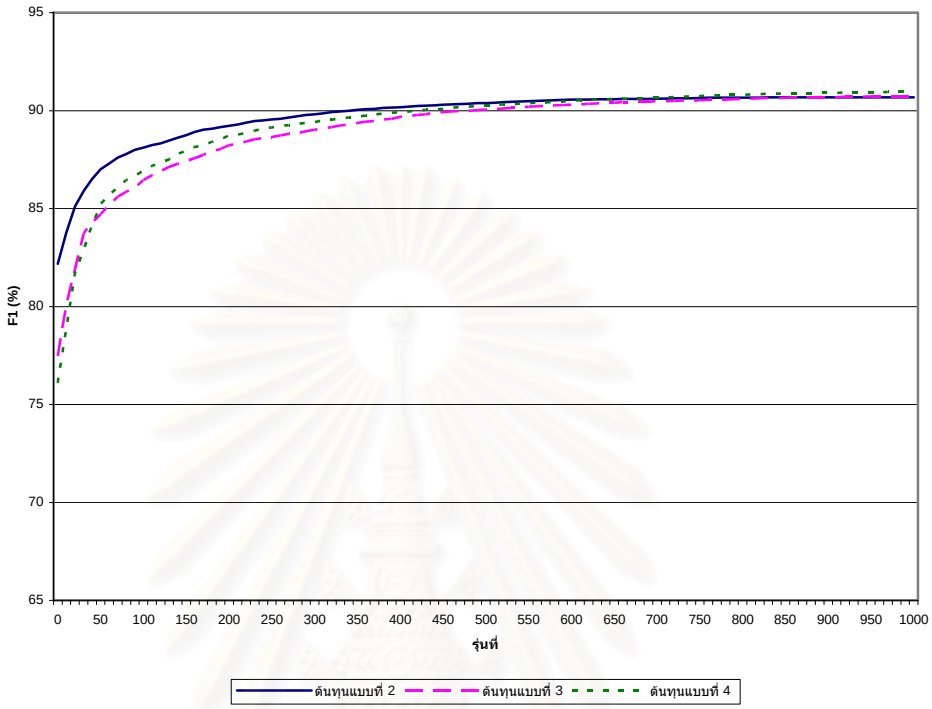
ตารางที่ 5.7 ผลการค้นคืนกรณีคำอังกฤษทับศัพท์คำไทย เมื่อให้ค่าต้นทุนเป็นแบบต่างๆ และใช้การเข้ารหัสคำด้วยนิวรอลเน็ตเวิร์กและแบบจำลองฮิดเด็นมาร์คอฟ

ต้นทุนแบบที่	นิวรอลเน็ตเวิร์ก			แบบจำลองฮิดเด็นมาร์คอฟ		
	ค่าแม่นยำ	ค่าเรียกคืน	F1	ค่าแม่นยำ	ค่าเรียกคืน	F1
1	0.9943	0.6920	0.8159	0.9940	0.5833	0.7351
2	0.9722	0.9435	0.9576	0.9623	0.7000	0.8104
3	0.9661	0.9472	0.9565	0.9598	0.7020	0.8107
4	0.9702	0.9420	0.9559	0.9606	0.7033	0.8119

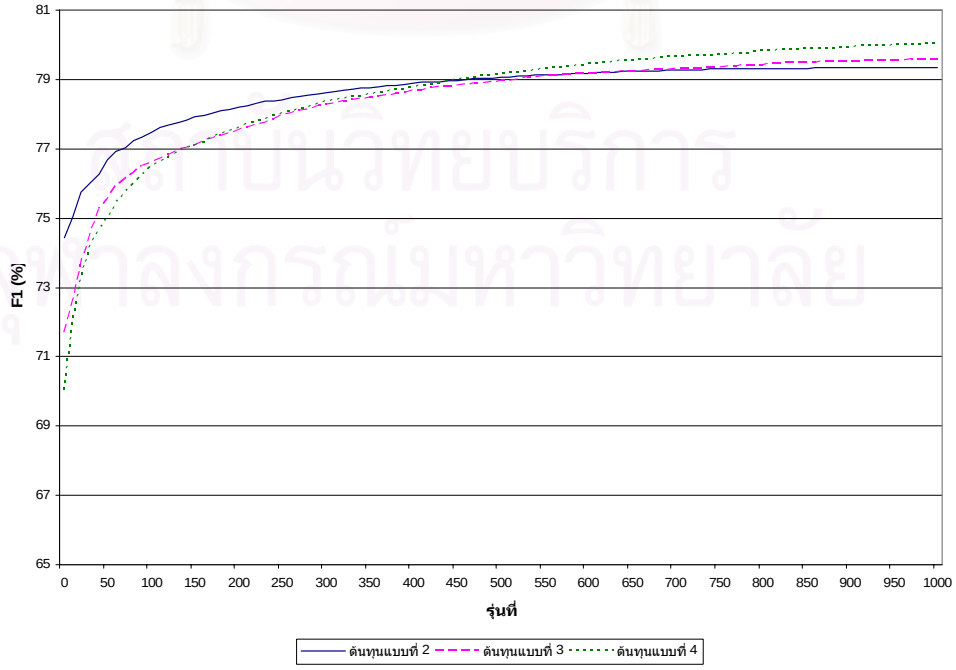
5.6 ขั้นตอนวิธีเชิงพันธุกรรมกับงานวิจัยของประยุทธ สุวรรณวิสารท

ในงานวิจัยเดิมของประยุทธ [11] เฉพาะสำหรับการค้นคืนคำอังกฤษทับศัพท์คำไทยนั้นได้เสนอการใช้ต้นทุนการแก้ไขอักขระหลายระดับขึ้นเพื่อใช้ในการเปรียบเทียบรหัสคำ และกำหนดหลักเกณฑ์ขึ้นมาสำหรับกำหนดค่าต้นทุนในตารางต้นทุนการแก้ไขอักขระ ในการทดลองนี้จึงได้นำเอางานวิจัยนั้นมาประยุกต์โดยเพิ่มการใช้ขั้นตอนวิธีเชิงพันธุกรรมในการหาตารางต้นทุนการแก้ไขอักขระและใช้ชุดค่าต้นทุนการแทนที่อักขระเช่นเดียวกับของประยุทธ ส่วนวิธีการเข้ารหัสคำนั้นใช้ขั้นตอนวิธีของประยุทธ ตารางที่ 5.8 แสดงผลการทดลองโดยเปรียบเทียบการใช้ตารางต้นทุนการแก้ไขอักขระจากงานวิจัยของประยุทธ และตารางต้นทุนการแก้ไขอักขระที่ได้จาก

ขั้นตอนวิธีเชิงพันธุกรรม ซึ่งจะเห็นได้ว่าได้ผลการค้นคืนที่ใกล้เคียงกันโดยการใช้ต้นทุนการแก้ไขอักขระของประยุทธได้ผลการค้นคืน F1 เป็น 0.9100 ส่วนการใช้ต้นทุนการ



รูปที่ 5.5 ผลการค้นคืนคำอังกฤษทับศัพท์คำไทยเมื่อใช้โครโมโซมที่ดีที่สุดในแต่ละรุ่นของขั้นตอนวิธีเชิงพันธุกรรมสำหรับการเข้ารหัสคำด้วยนิวรอลเน็ตเวิร์ก

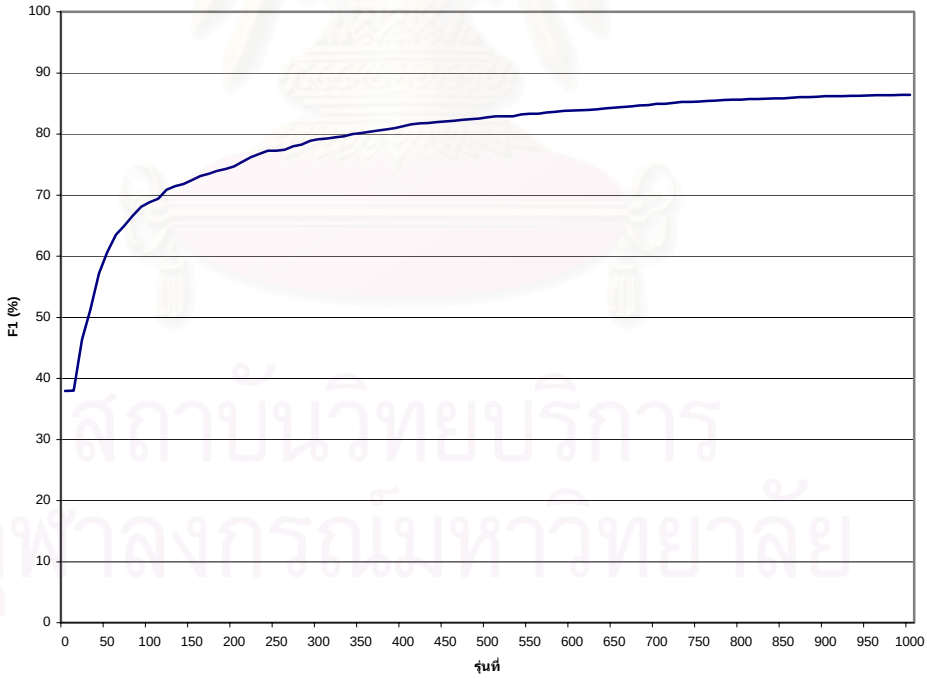


รูปที่ 5.6 ผลการค้นคืนคำอังกฤษทับศัพท์คำไทยเมื่อใช้โครโมโซมที่ดีที่สุดในแต่ละรุ่นของขั้นตอนวิธีเชิงพันธุกรรมสำหรับการเข้ารหัสคำด้วยแบบจำลองฮิดเด็นมาร์คอฟ

แก้ไขอักขระจากขั้นตอนวิธีเชิงพันธุกรรมจะได้ผลการค้นคืน F1 เป็น 0.9087 รูปที่ 5.7 แสดงผลการค้นคืนเฉลี่ยเมื่อใช้โครโมโซมที่ดีที่สุดในแต่ละรุ่นของขั้นตอนวิธีเชิงพันธุกรรม

ตารางที่ 5.8 ผลการค้นคืนคำอังกฤษทับศัพท์คำไทยเมื่อใช้ตารางต้นทุนการแก้ไขอักขระจากงานวิจัยของประยุทธ และจากการใช้ขั้นตอนวิธีเชิงพันธุกรรม

ใช้ตารางต้นทุนของประยุทธ			ใช้ตารางต้นทุนจากขั้นตอนวิธีเชิงพันธุกรรม		
ค่าแม่นยำ	ค่าเรียกคืน	F1	ค่าแม่นยำ	ค่าเรียกคืน	F1
0.9796	0.8496	0.9100	0.9397	0.8798	0.9087



รูปที่ 5.7 ผลการค้นคืนคำอังกฤษทับศัพท์คำไทยเมื่อใช้โครโมโซมที่ดีที่สุดในแต่ละรุ่นของขั้นตอนวิธีเชิงพันธุกรรมสำหรับการดัดแปลงงานวิจัยของประยุทธ สุวรรณวิสาทร

5.7 วิเคราะห์ผลการทดลองการค้นคืน

ผลการทดลองการค้นคืนที่ผ่านมาแสดงให้เห็นว่า เมื่อใช้ขั้นตอนวิธีเชิงพันธุกรรมในการหาตารางต้นทุนการแก้ไขอักขระที่เหมาะสมโดยใช้ชุดค่าต้นทุนหลายระดับสามารถให้ผลการค้นคืนที่ดีกว่าการใช้ต้นทุนการแก้ไขอักขระที่กำหนดต้นทุนเพียงระดับเดียว ประเด็นต่อมาคือความแตกต่างของวิธีการเข้ารหัสคำแบบต่างๆ ตารางที่ 5.9 เปรียบเทียบผลการค้นคืนคำไทยทับศัพท์คำอังกฤษจากแต่ละการทดลองที่ผ่านมา โดยนำผลที่ดีที่สุดมาแสดง แบ่งตามวิธีการเข้ารหัสคำและวิธีการหาต้นทุนการแก้ไขอักขระ จะเห็นได้ว่าเมื่อมีการใช้ขั้นตอนวิธีเชิงพันธุกรรมในการหาต้นทุนการแก้ไขอักขระ การใช้นิวรอลเน็ตเวิร์กในการเข้ารหัสคำให้ผลการค้นคืนที่ดีกว่าการใช้แบบจำลองฮิดเด็นมาร์คอฟ ส่วนตารางที่ 5.10 เปรียบเทียบกรณีคำอังกฤษทับศัพท์คำไทย ซึ่งแสดงให้เห็นอีกเช่นกันว่าการเข้ารหัสคำด้วยนิวรอลเน็ตเวิร์กให้ผลการค้นคืนที่ดีกว่าการใช้ฮิดเด็นมาร์คอฟรวมทั้งวิธีการของประยุทธอีกด้วย

ตารางที่ 5.9 เปรียบเทียบผลการค้นคืนคำไทยทับศัพท์คำอังกฤษด้วยวิธีการเข้ารหัสคำและการหาต้นทุนการแก้ไขอักขระต่างๆ

วิธีการ		ค่าแม่นยำ	ค่าเรียกคืน	F1
เข้ารหัสคำ	ต้นทุนแก้ไขอักขระ			
นิวรอลเน็ตเวิร์ก	ขั้นตอนวิธีเชิงพันธุกรรม	0.9635	0.8730	0.9160
ฮิดเด็นมาร์คอฟ	ขั้นตอนวิธีเชิงพันธุกรรม	0.9583	0.6759	0.7926

ตารางที่ 5.10 เปรียบเทียบผลการค้นคืนคำอังกฤษทับศัพท์คำไทยด้วยวิธีการเข้ารหัสคำและการหาต้นทุนการแก้ไขอักขระต่างๆ

วิธีการ		ค่าแม่นยำ	ค่าเรียกคืน	F1
เข้ารหัสคำ	ต้นทุนแก้ไขอักขระ			
ประยุทธ	ประยุทธ	0.9796	0.8496	0.9100
ประยุทธ	ขั้นตอนวิธีเชิงพันธุกรรม	0.9397	0.8798	0.9087
นิวรอลเน็ตเวิร์ก	ขั้นตอนวิธีเชิงพันธุกรรม	0.9722	0.9435	0.9576
ฮิดเด็นมาร์คอฟ	ขั้นตอนวิธีเชิงพันธุกรรม	0.9606	0.7033	0.8119

5.8 สรุป

ในบทนี้ได้กล่าวถึงผลการทดลองในการใช้นิวรอลเน็ตเวิร์กและแบบจำลองฮิดเด็นมาร์คอฟในการเข้ารหัสคำ และการใช้ขั้นตอนวิธีเชิงพันธุกรรมในการค้นคืนข้ามภาษา สำหรับนิวรอลเน็ตเวิร์กนั้นได้แสดงผลความถูกต้องเมื่อใช้จำนวนนิวรอนในชั้นซ่อนต่างๆกัน และสำหรับแบบจำลองฮิดเด็นมาร์คอฟนั้นได้ทดลองใช้จำนวนสถานะและอันดับต่างๆกัน เพื่อสร้างตัวเข้ารหัสคำที่ให้ค่าความถูกต้องสูงสุด หลังจากนั้นได้แสดงผลการทดลองการค้นคืนเมื่อใช้ขั้นตอนวิธีเชิงพันธุกรรมในการหาตารางต้นทุนการแก้ไขอักขระที่เหมาะสมเพื่อเพิ่มประสิทธิภาพของการค้นคืนให้มากขึ้น โดยได้ทดลองกับชุดค่าต้นทุนแบบต่างๆซึ่งมีหลายระดับ และข้อมูลคำไทยทับศัพท์คำอังกฤษและข้อมูลคำอังกฤษทับศัพท์คำไทย พบว่าสามารถให้ผลการค้นคืนที่ดีกว่าการใช้ต้นทุนการแก้ไขอักขระที่กำหนดต้นทุนเพียงระดับเดียว นอกจากนี้การเปรียบเทียบกันแสดงให้เห็นว่าเมื่อใช้ขั้นตอนวิธีเชิงพันธุกรรมในการหาต้นทุนการแก้ไขอักขระแล้วการใช้นิวรอลเน็ตเวิร์กในการเข้ารหัสคำให้ผลการค้นคืนที่ดีกว่าการใช้แบบจำลองฮิดเด็นมาร์คอฟ และวิธีการของประยุทธ

บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

6.1 สรุปผลการวิจัย

งานวิจัยนี้เสนอการใช้นิวรอลเน็ตเวิร์ก แบบจำลองฮิดเด็นมาร์คอฟ และขั้นตอนวิธีเชิงพันธุกรรม เพื่อการค้นคืนข้ามภาษาสำหรับคำทับศัพท์ภาษาไทย-อังกฤษ ในการค้นคืนนั้นอาศัยรหัสคำซึ่งแทนเสียงอ่านของคำในการเปรียบเทียบคำแต่ละคู่ว่ามีเสียงอ่านตรงกันหรือไม่ ในงานวิจัยนี้ใช้วิธีการนิวรอลเน็ตเวิร์ก และแบบจำลองฮิดเด็นมาร์คอฟในการเข้ารหัสคำ เนื่องจากปัญหาการเข้ารหัสคำนั้นสามารถจัดว่าเป็นปัญหาการจำแนกประเภทได้ สำหรับการเปรียบเทียบรหัสคำในการค้นคืนนั้น ใช้วิธีเปรียบเทียบเชิงประมาณด้วยขั้นตอนวิธีระยะแก้ไขสั้นสุด ซึ่งในขั้นตอนวิธีนี้เราสามารถกำหนดต้นทุนการแก้ไขอักขระได้หลายระดับ แต่การกำหนดต้นทุนการแก้ไขอักขระแบบนี้ทำให้เกิดปัญหาในการกำหนดค่าที่เหมาะสม ดังนั้นงานวิจัยนี้จึงได้ใช้ขั้นตอนวิธีเชิงพันธุกรรมเพื่อคำนวณต้นทุนการแก้ไขอักขระ ในการทดลองได้แบ่งออกเป็นการทดลองสำหรับกรณีคำไทยทับศัพท์คำอังกฤษ และกรณีคำอังกฤษทับศัพท์คำไทย แต่ละชุดแบ่งการทดลองตามวิธีการเข้ารหัสคำคือ นิวรอลเน็ตเวิร์กและแบบจำลองฮิดเด็นมาร์คอฟ นอกจากนี้ยังแบ่งการทดลองตามการกำหนดค่าต้นทุนการแก้ไขอักขระอีกด้วย จากผลการทดลองพบว่าได้ผลการค้นคืน F1 ประมาณ 91-95% เมื่อใช้นิวรอลเน็ตเวิร์กสำหรับเข้ารหัสคำ และประมาณ 79-81% เมื่อใช้แบบจำลองฮิดเด็นมาร์คอฟสำหรับการเข้ารหัสคำ ซึ่งจากทุกการทดลองพบว่าการใช้ตารางต้นทุนการแก้ไขอักขระจากขั้นตอนวิธีเชิงพันธุกรรมให้ผลการค้นคืนที่ดีกว่าการกำหนดให้ต้นทุนการแก้ไขอักขระมีระดับเดียว นอกจากนี้ยังได้นำงานวิจัยของ ประยุทธ สุวรรณวิสาทร มาดัดแปลงโดยเพิ่มการใช้ขั้นตอนวิธีเชิงพันธุกรรมเข้าไปด้วย จากการทดลองพบว่าสามารถให้ผลการค้นคืนประมาณ 91% ซึ่งใกล้เคียงกับผลการค้นคืนเมื่อใช้ต้นทุนการแก้ไขอักขระของประยุทธ เมื่อเปรียบเทียบกันแล้วพบว่าการใช้นิวรอลเน็ตเวิร์กร่วมกับขั้นตอนวิธีเชิงพันธุกรรมในการหาตารางต้นทุนการแก้ไขอักขระให้ผลการค้นคืนได้ดีที่สุด เมื่อเทียบกับการเข้ารหัสคำด้วยฮิดเด็นมาร์คอฟและวิธีการของประยุทธ

6.2 ข้อเสนอแนะ

1. หากข้อมูลคำทับศัพท์ที่ใช้ในการทดลองมีจำนวนและความหลากหลายมากกว่านี้ จะให้ผลการทดลองดียิ่งขึ้น

2. การฝึกแบบจำลองฮิดเด็นมาร์คอฟนั้นในงานวิจัยนี้ใช้ขั้นตอนวิธี Baum-welch แต่ยังมีวิธีอื่นอีกเช่น วิธี Maximum Mutual Information (MMI)
3. การทดลองในส่วนของขั้นตอนวิธีเชิงพันธุกรรมใช้เวลาค่อนข้างมากในการประมวลผล ดังนั้นอาจต้องใช้เทคนิคการโปรแกรมเชิงขนาน (Parallel programming) หรือ การคำนวณแบบกระจาย (Distributed computing) มาช่วยให้การประมวลผลใช้เวลาน้อยลง
4. สำหรับคำไทยนั้นมีการใช้สระลดรูป ได้แก่ –ะ โ-ะ และ –อ ดังนั้นเราสามารถสร้างขั้นตอนวิธีขึ้นมาสำหรับจัดการกับปัญหานี้โดยเฉพาะได้ เพื่อให้สามารถจำแนกได้ชัดเจนว่าตำแหน่งใดของคำที่มีการใช้สระลดรูป
5. การใช้วิธีการแบ่งพยางค์อาจช่วยให้การจำแนกรหัสเสียงดีขึ้นได้ เพราะจะทำให้เราทราบได้ว่าตัวอักษรในพยางค์เป็นพยัญชนะต้นหรือตัวสะกด ซึ่งลดความกำกวมเนื่องจากการที่ตัวอักษรบางตัวมีรหัสเสียงทั้งกรณีที่เป็นพยัญชนะต้นและกรณีที่เป็นตัวสะกด

รายการอ้างอิง

1. อุไรรัตน์ บุญภานนท์. การถอดอักษรภาษาอังกฤษเป็นไทยโดยใช้หลักภาษาศาสตร์. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาภาษาศาสตร์ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิยาลัย, 2526.
2. Michell, T. M. Machine Learning. The McGraw–Hill Companies, 1997.
3. Rabiner, L., and Juang, B. Fundamentals of Speech Recognition. New Jersy: Prentice-Hall, 1993.
4. Goldberg, D. E. Genetic Algorithm in Search, Optimization, and Machine Learning. Addison Wesley, 1989.
5. Frakes, W.B., and Baeza Yates, R. Information Retrieval : Data Structures & Algorithms. Englewood Cliffs, N.J.: Prentice Hall, 1992.
6. van Rijsbergen, C.J. Information Retrieval. Butterworths, London, 1979.
7. Zobel, J., and Dart, P. Phonetic String Matching: Lessons from Information Retrieval. In Proc. of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1996.
8. Binstock, A., and Rex, J. Practical Algorithms for Programmers. New York: Addison Wesley, 1995,
9. วรรณี อุดมพาณิชย์. การใช้หลักคำพ้องเสียง เพื่อค้นหาชุดอักขระภาษาไทยที่ออกเสียงเหมือนกัน. วิทยานิพนธ์ปริญญามหาบัณฑิต ภาควิชาภาษาศาสตร์ บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิยาลัย, 2526.
10. นิลเนตร อรุณวงศ์ ณ อยุธยา. การเปลี่ยนอักขระของคำในภาษาไทย โดยใช้หลักการของซาวด์เด็กซ์. ปริญญานิพนธ์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิยาลัย, 2534.
11. Suwanvisat P., and Prasitjutrakul, S. Thai-English Cross-Language Transliterated Word Retrieval using Soundex Technique. In Proc. of the National Computer Science and Engineering Conference 1998, Bangkok Thailand, Aug. 19-21, 1998.
12. Suwanvisat, P., and Prasitjutrakul, S. Transliterated Word Encoding and Retrieval Algorithms for Thai–English Cross–Language Retrieval. In Proc. of the National Computer Science and Engineering Conference 1999, Bangkok Thailand, Dec. 16–17, 1999.

13. ทัศนวรรณ ศูนย์กลาง, สมชาย ประสิทธิ์จูตระกูล, และบุญเสริม กิจศิริกุล. การเข้ารหัสคำทับศัพท์ภาษาไทย/อังกฤษเพื่อการค้นคืนข้ามภาษาด้วยเทคนิคนิวรอลเน็ตเวิร์ก. ในรายงานการประชุมวิชาการทางวิทยาศาสตร์และวิศวกรรมคอมพิวเตอร์แห่งชาติ ครั้งที่ 7 (NCSEC 2003), กรุงเทพฯ, ประเทศไทย, 16-17 ธันวาคม, 2543.
14. Duangpanyasawang T., and Kijsirikul, B. Combining Hidden Markov Models and Phonetic Trigrams in a Thai Soundex System. In The 2001 International Conference on Information Technology for the New Millennium (IConIT 2001), Bangkok, 28-30 May 2001.
15. หลักเกณฑ์การถอดอักษรไทยเป็นอักษรโรมันแบบถ่ายเสียง. กรุงเทพมหานคร: ราชบัณฑิตยสถาน, 2542.
16. หลักเกณฑ์การทับศัพท์ภาษาอังกฤษ ฉบับราชบัณฑิตยสถาน. กรุงเทพมหานคร: ราชบัณฑิตยสถาน, 2532.
17. อุปกิตศิลปสาร, พระยา. หลักภาษาไทย. พิมพ์ครั้งที่ 11. กรุงเทพมหานคร: สำนักพิมพ์ไทยวัฒนาพานิช, 2545.

ภาคผนวก

ภาคผนวก ก
การใช้อักษรโรมันแทนอักขระไทย

ตารางที่ ก.1 การใช้อักษรโรมันแทนพยัญชนะไทยของราชบัณฑิตยสถาน

พยัญชนะไทย	อักษรโรมัน	
	พยัญชนะต้น	ตัวสะกด
ก	K	K
ข	KH	K
ฃ	KH	K
ค	KH	K
ฅ	KH	K
ฆ	KH	K
ง	NG	NG
จ	CH	T
ฉ	CH	T
ช	CH	T
ซ	S	T
ฌ	CH	T
ญ	Y	N
ฎ	D	T
ฏ	T	T
ฐ	TH	T
ฑ	D	T
ฒ	TH	T
ณ	TH	T
ณ	N	N
ด	D	T
ต	T	T
ถ	TH	T

พยัญชนะไทย	อักษรโรมัน	
	พยัญชนะต้น	ตัวสะกด
ธ	TH	T
น	N	N
บ	B	P
ป	P	P
ผ	PH	P
ฝ	F	P
พ	PH	P
ฟ	F	P
ภ	PH	P
ม	M	M
ย	Y	-
ร	R	N
ฤ	R	R
ล	L	N
ฦ	L	L
ว	W	-
ศ	S	T
ษ	S	T
ส	S	T
ห	H	-
ฬ	L	N
อ	H	-
ฮ	H	-

ตารางที่ ก.1 (ต่อ) การใช้อักษรโรมันแทนพยัญชนะไทยของราชบัณฑิตยสถาน

พยัญชนะไทย	อักษรโรมัน	
	พยัญชนะต้น	ตัวสะกด
ท	TH	T

พยัญชนะไทย	อักษรโรมัน	
	พยัญชนะต้น	ตัวสะกด
ทร	S	T

ตารางที่ ก.2 การใช้อักษรโรมันแทนสระไทยของราชบัณฑิตยสถาน

สระไทย	อักษรโรมัน
ะ ั า	A
ำ	AM
ิ ี ิย	I
ึ ื ือ ุ ู	U
เ-ะ เ็ เ-	E
แ-ะ แ็ แ-	AE
โ-ะ โ- เ-าะ ็อ- -อ อ-	O
เ-อะ เ-อ เ ิ	OE
เ ียะ เ ีย	IA
เ ือะ เ ือ ัวะ ัว -ว-	UA
ใ- ไ- ไ-ย ัย -าย	AI
เ-า -าว	AO
ุย ึย	UI
โ-ย -อย	OI
ิว	IU
เ ็ว เ-ว	EO
เ-ย	OEI
เ ือย -วย	UAI
แ-ว	AEU
เ ียว	IEU

ภาคผนวก ข
หน่วยเสียงในภาษาไทยและภาษาอังกฤษ

หน่วยเสียงในภาษาไทย

ภาษาไทยมีหน่วยเสียงพยัญชนะ 21 หน่วยเสียง ดังตารางที่ ข.1 หน่วยเสียงสระ 21 หน่วยเสียง ดังตารางที่ ข.2 และหน่วยเสียงวรรณยุกต์ 5 หน่วยเสียงได้แก่ สามัญ เอก โท ตรี จัตวา

ตารางที่ ข.1 หน่วยเสียงพยัญชนะในภาษาไทย

ก	ช ศ ษ ส ทร	ณ น หน	ม หม
ข ฃ ค ฅ ฆ	ญ ย หย หญ	บ	ร
ง หง	ฎ ด ฑ	ป	ล ฬ หล
จ จร	ฏ ต	ผ พ ภ	ว หว
ฉ ช ฌ	ฐ ฑ ถ ท ธ	ฝ ฟ	ห ฮ
			อ

ตารางที่ ข.2 หน่วยเสียงพยัญสระในภาษาไทย

อิ	แอะ	เออะ	อุ	เอาะ	อัวะ อัว
อี	แอ	เออ	อู	ออ	
เอะ	อึ	อะ	โอะ	เอียะ เอีย	
เอ	อื	อา	โอ	เออะ เอือ	

ระบบเสียงในภาษาอังกฤษ

ภาษาอังกฤษมีหน่วยเสียงพยัญชนะ 24 หน่วยเสียง ดังตารางที่ ข.3 และหน่วยเสียงสระ 20 หน่วยเสียง ดังตารางที่ ข.4

ตารางที่ ข.3 หน่วยเสียงพยัญชนะในภาษาอังกฤษ

เสียง	คำตัวอย่าง	เสียง	คำตัวอย่าง	เสียง	คำตัวอย่าง
/ p /	pen	/ f /	fall	/ h /	how
/ b /	bad	/ v /	voice	/ m /	man
/ t /	tea	/ θ /	thin	/ n /	no
/ d /	did	/ ð /	then	/ ŋ /	sing
/ k /	cat	/ s /	so	/ l /	leg
/ g /	got	/ z /	zoo	/ r /	red
/ tʃ /	chin	/ ʃ /	she	/ j /	yes
/ dʒ /	jam	/ ʒ /	vision	/ w /	wet

ตารางที่ ข.4 หน่วยเสียงสระในภาษาอังกฤษ

เสียง	คำตัวอย่าง	เสียง	คำตัวอย่าง	เสียง	คำตัวอย่าง
/ i: /	see	/ ʊ /	put	/ aI /	tie
/ I /	sit	/ u: /	too	/ aʊ /	now
/ e /	ten	/ ʌ /	cup	/ ɔI /	join
/ æ /	hat	/ ɜ: /	fur	/ Iə /	near
/ a: /	arm	/ ə /	ago	/ eə /	hair
/ o /	got	/ eI /	page	/ ʊə /	tour
/ ɔ: /	saw	/ əʊ /	home		

ภาคผนวก ค
ตัวอย่างข้อมูลคำทับศัพท์ที่ใช้ในงานวิจัย

ตัวอย่างคำอังกฤษและคำไทยทับศัพท์คำอังกฤษ

Liberty	ลิเบอร์ตี	gyro	ไจโร	gowland	เกาว์แลนด์
europe	ยุโรป	berkelium	เบอร์คีเลียม	anderson	แอนเดอร์สัน
aurora	ออโรรา	juice	จูซ	frisch	ฟรีซ
cytosol	ไซโทซอล	lothar	โลทาร์	bowman	โบว์แมน
playfair	เพลย์แฟร์	collenchyma	คอลเลงคิมา	helium	ฮีเลียม
zeta	ซีตา	boson	โบซอน	stokes	สโตกส์
iodide	ไอโอไดด์	einstein	ไอน์สไตน์	marconi	มาร์โคนี
okhotsk	โอค็อตสก์	chrysler	ไครส์เลอร์	warren	วาร์เรน
maclagan	แมกลาเกน	gaussian	เกาส์เซียน	delta	เดลต้า
rye	ไรย์	mil	มิล	allantois	แอลแลนทอยส์
arc	อาร์ก	broadway	บรอดเวย์	sikh	ซิก
mozambique	โมซัมบิก	thomson	ทอมสัน	hankel	ฮันเกล
suzuki	ซูซูกิ	romer	เรอเมอร์	micelle	ไมเซลล์
lantis	แลนทิส	peta	เพตะ	factor	แฟกเตอร์
dewey	ดิวอีย์	klein	ไคลน์	zygomata	ไซโกมาตา
cotangent	โคแทนเจนต์	manganese	แมงกานีส	karl	คาร์ล
cosecant	โคเซแคนต์	golf	กอล์ฟ	bract	แบร็กต์
chromosphere	โครโมสเฟียร์	harsh	ฮาร์ช	joule	จูล

ตัวอย่างคำไทยและคำอังกฤษทับศัพท์คำไทย

weerachai	วีรชัย	kijluakiat	กิจลือเกียรติ	worawas	วรวัสส์
plianrungsi	เปลี่ยนรังษี	puttakrong	พุทธกรอง	tongchai	ทองชัย
vilaiphan	วิไลพันธุ์	rossukon	รสสุคนธ์	wasana	วาสนา
sukhaboon	สุขาบูรณ์	mitisubin	มิติสุบิน	achaporn	อัชพร
prangsiri	ปรางศิริ	veerasak	วีรศักดิ์	poonpissamai	พูนพิศมัย
puengrusme	พึงรัศมี	kanokrat	กนกรัตน์	wisanupong	วิษณุพงษ์
intapuntee	อินทปันติ	sarakit	ศารกิจ	sapatporn	สภัทร์พร
onumpai	อรอำไพ	ekarat	เอกรัฐ	silarujisun	ศิลารุจิสรรค์
muendech	หมื่นเดช	siriphap	ศิริภาพ	wilailak	วิไลลักษณ์
chatkaew	ฉัตรแก้ว	khrongwong	ครองวงศ์	wichian	วิเชียร
ruethai	ฤทัย	rapeephan	รพีพรรณ	marianukroh	มารีอนุเคราะห์
ratree	ราตรี	oranuch	อรนุช	warit	วริษฐ์
jittima	จิตติมา	sawart	สวาท	pisithsak	พิสิษฐ์ศักดิ์
phamornthep	ภมรเทพ	tiawsirisup	เตียวศิริทรัพย์	laosunthara	เหล่าสุนทร
saowarat	เสาวรัตน์	charanvas	จรัญวาศน์	suproongruing	ทรัพย์รุ่งเรือง
manoon	มนูญ	kanitta	ขนิษฐา	nattaphan	ณัฐพันธุ์
keng	เก่ง	phongphew	ผ่องแผ้ว	somjate	สมเจตน์
limlawan	ลิมป์ลาวัณย์	weerasak	วีระศักดิ์	leelawat	ลีละวัฒน์
sudarat	สุดารัตน์	chumpot	จุมภฏ	kuankid	ควรคิด
suthip	สุทิพย์	charinee	ฉาริณีย์	pongput	ผองพุทธ
natakit	ณัฐกิตติ์	puttipong	พุทธิพงศ์	thansathit	ตัณสถิตย์

ประวัติผู้เขียนวิทยานิพนธ์

นายศิริพจน์ สุรบถโสภณ เกิดเมื่อวันที่ 15 พฤษภาคม พ.ศ. 2522 ที่กรุงเทพมหานคร สำเร็จการศึกษาระดับปริญญาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อปี พ.ศ. 2543 หลังจากนั้นได้ทำงานเป็นวิศวกรทางด้านเทคโนโลยีสารสนเทศที่บริษัทสตรีม ไอที คอนซัลติง จำกัด เป็นเวลา 2 ปี ต่อมาได้เข้าศึกษาต่อในหลักสูตรวิศวกรรมศาสตรมหาบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อพ.ศ. 2545 มีผลงานทางวิชาการคือบทความเรื่อง “การค้นคืนข้ามภาษาสำหรับคำทับศัพท์ภาษาไทย/อังกฤษด้วยวิธีการนิวรอลเน็ตเวิร์กและขั้นตอนวิธีเชิงพันธุกรรม” (Thai-English Cross-Language Transliterated Word Retrieval Using Neural Networks and Genetic Algorithms) ซึ่งได้รับการตีพิมพ์และนำเสนอในงานประชุมวิชาการวิทยาการคอมพิวเตอร์และวิศวกรรมคอมพิวเตอร์แห่งชาติ 2546 (The National Computer Science and Engineering Conference: NCSEC’03) เมื่อวันที่ 28-30 ตุลาคม พ.ศ. 2546