บทที่ 2

งานวิจัยที่เกี่ยวข้อง

เนื่องจากการตัดคำได้มีการพัฒนาติดต่อกันมาเป็นเวลายาวนาน ทำให้มีงานวิจัยด้านการตัดคำเกิดขึ้นมากมายหลายวิธี ซึ่งสามารถแบ่งงานวิจัยเหล่านี้ออกตามวิธีหลักที่ใช้ในการตัดคำได้เป็นดังนี้คือ 1. วิธีการใช้กฎ 2. วิธีการใช้พจนานุกรม 3. วิธีการใช้คลังข้อความ

วิธีการใช้กฎ

งานวิจัยรุ่นแรกๆได้นำวิธีการใช้กฎมาใช้ในการตัดพยางค์ ซึ่งได้มีการพัฒนากฎที่ใช้ในการตัดพยางค์ดังในงานวิจัยต่อไปนี้

- งานวิจัยของ ยุพิน ไทยรัตนานนท์ [21] เป็นงานวิจัยการตัดพยางค์โดยการใช้กฎที่สร้างขึ้นจากหลักไวยากรณ์ภาษาไทย และมีการจัดเก็บพยางค์ต่างๆที่เป็นข้อยกเว้นไว้ในแฟ้มข้อมูล เนื่องจากมีบางพยางค์ไม่เป็นไปตามกฎที่สร้างไว้

ลักษณะของกฎที่นำมาใช้ในการตัดพยางค์ภายในงานวิจัยนี้ ได้สร้างมาจากลักษณะไวยากรณ์ทางภาษาไทย โดยมีการพิจารณาจากลักษณะของอักษรที่ปรากฏในพยางค์หรือคำ ซึ่งทำให้มีการจัดหมวดหมู่ตัวอักษรภาษาไทย โดยการแบ่งหมวดตามการนำไปใช้ ซึ่งสามารถแบ่งได้เป็น 5 กลุ่มใหญ่ๆ ดังต่อไปนี้ คือ

1. กลุ่มพยัญชนะ (Consonant)
    - พยัญชนะที่อยู่หน้าพยางค์เสมอ
    - พยัญชนะที่ส่วนใหญ่จะอยู่หน้าพยางค์
    - พยัญชนะที่เป็นตัวสะกด
    - พยัญชนะที่เป็นสระ
    - อื่นๆ
2. กลุ่มสระ (Vowel)
    - สระที่ไม่ต้องมีตัวสะกด
    - สระที่จะอยู่หน้าพยางค์เสมอ
    - สระที่ส่วนใหญ่จะมีตัวสะกดร่วมด้วย
    - สระที่มีหรือไม่มีตัวสะกดร่วมด้วย
3. กลุ่มวรรณยุกต์ (Tone mark)

4. กลุ่มตัวเลข (Numeral)

5. กลุ่มอักษรพิเศษ (Special character)

ขั้นตอนการทำงานของวิธีการนี้จะตัดพยางค์จากขวามาซ้าย โดยใช้กฎต่างๆ ที่สร้างขึ้นมาจากลักษณะของตัวอักษรดังที่ได้กล่าวไปแล้ว และกฎต่างๆ ที่สร้างขึ้นมานั้นจะจัดเก็บไว้ภายในรหัสต้นฉบับ (Source code) ซึ่งทำให้การเพิ่มหรือแก้ไขกฎไม่สามารถทำได้สะดวก และจากการทดสอบปรากฏว่าผลลัพธ์ที่ได้จากการตัดพยางค์ด้วยวิธีการนี้ จะได้ผลความถูกต้องไม่น้อยกว่า 85%

- งานวิจัยของ สุรินทร์ จรรยาพรพงษ์ [10] เป็นงานวิจัยเกี่ยวกับการตัดพยางค์ภาษาไทยโดยใช้กฎ โดยกฎที่นำมาใช้นั้นได้นำมาจากหลักไวยากรณ์ภาษาไทย และได้ทำการวิเคราะห์ลักษณะต่างๆ ของพยางค์ภาษาไทย โดยลักษณะของกฎที่ได้นี้สามารถแบ่งได้เป็น 2 ชนิดคือ กฎการหาขอบเขตหน้า (Front boundary recognition rule) และ กฎการหาขอบเขตหลัง (Tail boundary recognition rule) และในแต่ละกฎยังแบ่งออกเป็น 2 กลุ่มย่อยๆ คือแบ่งตามคุณสมบัติของตัวอักษรโดยกฎที่ได้ออกมานี้จะจัดให้อยู่ในกลุ่มเอ (Group A) และแบ่งตามคุณสมบัติของรูปแบบการใช้สระแต่ละตัวซึ่งกฎที่ได้ออกมานี้จะแบ่งให้อยู่ในกลุ่มบี (Group B)

ตัวอย่างกฎที่ได้จากคุณสมบัติของอักษรในการหาขอบเขตหน้าของพยางค์ เช่น

กฎ A-1F : สระต่างๆ เหล่านี้ ะ า [illegible] และวรรณยุกต์ [illegible] จะต้องมีพยัญชนะอยู่ข้างหน้าอย่างน้อย 1 ตัวอักษรเสมอ

กฎ A-2F : สระ เ แ ไ โ ส่วนใหญ่จะเป็นตัวอักษรแรกในพยางค์เสมอ ยกเว้นคำบางคำ ตัวอย่างเช่น ขโมย ชโลม ทแยง อเนก สไบ ซไม เป็นต้น

กฎ A-3F : สระ ใ จะเป็นตัวอักษรแรกของพยางค์เสมอ โดยไม่มีข้อยกเว้น

ตัวอย่างกฎที่ได้จากคุณสมบัติของอักษรในการหาขอบเขตหลังของพยางค์ เช่น

กฎ A-1T : พยัญชนะต่อไปนี้ ศ ณ ญ ษ ฐ ฎ ฏ ฒ ฬ ฌ จะเป็นตัวสะกดเสมอ ยกเว้นพยางค์ดังต่อไปนี้ ศก ศร ญวน ศตวรรษ และยังมีพยางค์อื่นๆ อีก แต่พยางค์เหล่านั้นสามารถจัดการได้โดยใช้ กฎ A-1F

กฎ A-2T : สระ ึ จะตัวมีตัวสะกดหนึ่งตัวเสมอ ยกเว้น พยางค์เหล่านี้ ฤ ฬ ฮึ

กฎ A-3T : ไม้หันอากาศ (ั) จะต้องตามด้วยตัวสะกดอย่างน้อย 1 ตัวอักษรเสมอ

นอกจากนี้ในงานวิจัยนี้ยังมีการสร้างกฎที่ใช้ในการแบ่งพยางค์ โดยกฎที่ได้จะพิจารณาจากคุณสมบัติของรูปแบบการใช้สระแต่ละตัวหรือกฎกลุ่มบี ซึ่งจะพิจารณาจากลักษณะ

ของพยางค์ในภาษาไทย โดยจากที่สังเกตเห็นคือลักษณะของพยางค์ในภาษาไทยนั้นจะมีรูปแบบที่แน่นอนและตายตัว ทำให้ในงานวิจัยนี้มีการสร้างกฎต่างๆ โดยอ้างอิงจากรูปแบบของพยางค์ต่างๆ ที่ปรากฏในภาษาไทย

ตัวอย่างกฎที่ได้จากคุณสมบัติของรูปแบบการใช้สระ ในการหาขอบเขตหลังของพยางค์ เช่น

กฎ B-1T : สระเหล่านี้ ั ิ ึ ื ถ้ามีวรรณยุกต์แล้วจะต้องมีตัวสะกด 1 ตัวเสมอ

กฎ B-4T : สระ ในรูปแบบดังต่อไปนี้ ัว ัย ัวะ ือ เ-า เ-อ เ-าะ เ-ะ แ-ะ โ-ะ เ-ียะ เ-อะ ไม่ต้องการตัวสะกด ยกเว้นบางพยางค์ในรูปแบบดังนี้ เ–า– และ เ–อ– โดยเครื่องหมาย – แทนตัวพยัญชนะ 1 ตัวอักษร

งานวิจัยนี้ได้ทดลองกับเอกสารต่างๆ จำนวน 100 เล่ม โดยเอกสารนั้นได้นำมาจากเอกสารชนิดต่างๆเช่น หนังสือพิมพ์ ปรัชญา วิทยาศาสตร์ ศาสนา ภาษาศาสตร์ ฯลฯ และจากการทดสอบปรากฏว่าสามารถตัดพยางค์ได้ถูกต้องถึง 96%

<u>วิธีการใช้พจนานุกรม</u>

ในงานวิจัยแรกๆเป็นการตัดพยางค์โดยการใช้กฎเพื่อหาขอบเขตของพยางค์ ต่อมาเป็นงานวิจัยเกี่ยวกับการตัดคำซึ่งเป็นการหาขอบเขตของคำ โดยการหาขอบเขตของคำไม่สามารถหาได้จากการใช้การตัดพยางค์เพียงอย่างเดียวได้ เนื่องจากคำประกอบด้วยพยางค์ 1 พยางค์หรือหลายพยางค์ก็ได้เช่นคำว่า ห้อง เดิน สะพาน กระดาน เป็นต้น ทำให้ได้มีการคิดค้นหาวิธีการตัดคำโดยใช้พจนานุกรมร่วมกับการใช้กฎในการตัดคำหลายวิธีดังในงานวิจัยต่อไปนี้

- งานวิจัยของยืน ภู่วรวรรณ และวิวรรธน์ อิ่มอารมณ์ [4] เป็นงานวิจัยการแบ่งพยางค์ด้วยพจนานุกรม ซึ่งถือได้ว่าเป็นงานวิจัยงานแรกของการตัดพยางค์ที่มีการนำพจนานุกรมเข้ามาใช้ โดยจะจัดเก็บพยางค์ต่างๆ ไว้ในพจนานุกรม และมีการนำกฎไวยากรณ์ต่างๆเข้ามาช่วยในกรณีที่ไม่พบพยางค์ในพจนานุกรม

หลักการทำงานของวิธีการตัดพยางค์ด้วยพจนานุกรมนี้ก็คือ จะทำการตรวจสอบสายอักขระ (String) ที่เข้ามาจากซ้ายไปขวากับพยางค์ที่ได้เก็บไว้ในพจนานุกรม ในกรณีที่ทำการตรวจสอบแล้วปรากฏว่าพบพยางค์มากกว่า 1 พยางค์ในพจนานุกรม ก็ให้ทำการเลือกแบ่งพยางค์โดยเลือกพยางค์ที่ยาวที่สุดแล้วทำเครื่องหมายจุดย้อนกลับกับพยางค์ที่เหลือ แล้วก็ทำต่อไปเรื่อยๆ

จนจบสายอักขระ แต่ถ้าในกรณีที่เลือกพยางค์ที่ยาวที่สุดไปแล้ว ทำให้เกิดพยางค์ที่ไม่ปรากฏในพจนานุกรมก็ยอมให้มีการย้อนรอย (Back Tracking) กลับไปยังจุดย้อนกลับล่าสุดเลือกพยางค์ที่ยาวรองลงมาแทน ซึ่งวิธีการนี้จะเป็นที่รู้จักกันในชื่อ การตัดคำแบบเลือกคำที่ยาวที่สุด (Longest Matching)

งานวิจัยนี้ได้มีการเปรียบเทียบความรวดเร็วในการแบ่งพยางค์ ซึ่งสรุปผลได้ว่าเมื่อนำพจนานุกรมเข้ามาใช้ในการแบ่งพยางค์จะสามารถตัดพยางค์ได้รวดเร็วกว่าการใช้กฎ โดยที่ความถูกต้องของการตัดพยางค์นั้นสามารถตัดได้ถูกต้องมากกว่า 99 % แต่สำหรับวิธีการนี้ก็ยังมีข้อเสียคือ ต้องเสียเนื้อที่ในการจัดเก็บพจนานุกรมในหน่วยความจำหลักเป็นจำนวน 50 กิโลไบต์

- งานวิจัยของ ดวงแก้ว สวามิภักดิ์ [1] งานวิจัยชิ้นนี้คือ การสร้างซอฟต์แวร์วิเคราะห์ไวยากรณ์ไทยภายใต้ระบบยูนิกซ์ เป็นงานวิจัยด้านการตัดคำภาษาไทยโดยใช้กฎทางไวยากรณ์ที่สร้างขึ้นมาเอง และมีการนำพจนานุกรมเข้ามาใช้ประกอบร่วมด้วย โดยสาเหตุที่นำทั้งกฎไวยากรณ์และพจนานุกรมเข้ามาช่วยในการตัดคำนั้น เพื่อที่จะแก้ไขปัญหาการตัดคำโดยใช้พจนานุกรมเพียงอย่างเดียว [4] ซึ่งไม่สามารถตัดคำได้ถูกต้องในกรณีที่คำนั้นไม่มีอยู่ในพจนานุกรม

งานวิจัยการตัดคำนี้มีการสร้างกฎต่างๆ ให้อยู่ในรูปแบบนิพจน์ที่มีกฎเกณฑ์ (Regular Expression) โดยกฎที่สร้างขึ้นมานี้ประกอบไปด้วย 43 กฎ (รายละเอียดของกฎต่างๆ สามารถดูได้ในภาคผนวก ข) ซึ่งกฎที่ได้มานี้จะไม่มีการรวมตัวสะกดเข้าไปในกฎด้วยยกเว้นบางกรณี เนื่องจากลักษณะของโปรแกรมเล็กซ์ (Lex) ภายใต้ระบบปฏิบัติการยูนิกซ์ที่ใช้ในงานวิจัยนี้จะพยายามสร้างกลุ่มตัวอักษร (Token) ที่มีขนาดที่ยาวที่สุดก่อน ดังนั้นถ้ามีการนำกฎที่มีตัวสะกดเข้ามาใช้ จะเป็นสาเหตุให้มีการรวมเอาอักษรตัวหน้าของคำถัดไปมาเป็นตัวสะกดได้ ซึ่งเมื่อได้ผ่านการวิเคราะห์ด้วยกฎแล้ว ขั้นตอนต่อไปก็จะมีการรวมกลุ่มตัวอักษรเข้าด้วยกัน โดยทำการตรวจสอบจากพจนานุกรม ส่วนโครงสร้างของพจนานุกรมใช้โครงสร้างข้อมูลแบบบีทรี (B-Tree) ที่มีฐานข้อมูลแบบรีเลชั่น (Relational DBMS) ซึ่งใช้คำเป็นดรรชนี (Index)

งานวิจัยนี้ได้แบ่งการวัดประสิทธิภาพของการตัดคำเป็น 2 ชนิดคือ 1. ความถูกต้องในเชิงของคำ และ 2. ความถูกต้องในเชิงของพยางค์ และได้ทดลองกับเอกสารจำนวน 17 ชนิด ซึ่งผลปรากฏว่าได้ความถูกต้องถึง 98.11% ในเชิงคำ และ 99.67% เชิงของพยางค์

- งานวิจัยของสัมพันธ์ ระรื่นรมย์ [8] เป็นงานวิจัยการแบ่งคำไทยด้วยพจนานุกรม โดยเป้าหมายของงานวิจัยนี้จะเน้นที่การเพิ่มประสิทธิภาพในด้านความรวดเร็วของขั้นตอนวิธีในการตัดคำ

และการลดขนาดของพจนานุกรม เนื่องจากเมื่อนำพจนานุกรมเข้ามาใช้ในการตัดคำแล้วจะทำให้ความถูกต้องในการตัดคำเพิ่มขึ้นมากกว่าการตัดคำใช้กฎอย่างเดียว ดังนั้นในงานวิจัยนี้จึงไม่ได้เน้นการเพิ่มประสิทธิภาพในด้านความถูกต้องมากนักเพราะถือว่าการตัดคำโดยใช้พจนานุกรมให้ค่าความถูกต้องที่สูงอยู่แล้ว

ขั้นตอนวิธีการตัดคำในงานวิจัยนี้จะคล้ายกับงานวิจัยการแบ่งพยางค์โดยใช้พจนานุกรม [4] คือใช้ขั้นตอนวิธีแบบเลือกคำที่ยาวที่สุดดังที่ได้กล่าวไปแล้ว แต่ในงานวิจัยนี้จะทำการจัดเก็บคำแทนพยางค์ในพจนานุกรม ตัวอย่างการตัดคำโดยเลือกคำที่ยาวที่สุดแสดงดังตารางที่ 2.1

ประโยค	คำที่ได้	คำที่ถูกเลือก
เขาหากระดาษ	เขา	เขา
หากระดาษ	หา, หาก	หาก
ระดาษ	-	(ย้อนรอย)
หากระดาษ	หา, หาก	หา (เลือกคำรองลงมา)
กระดาษ	กระดาษ	กระดาษ

ตารางที่ 2.1 ตารางแสดงการตัดคำแบบเลือกคำที่ยาวที่สุด

จากตารางที่ 2.1 จะแสดงการตัดคำแบบเลือกคำที่ยาวที่สุด โดยประโยคที่นำมาตัดคำคือ เขาหากระดาษ สามารถตัดคำได้เป็น เขา หา กระดาษ

ส่วนโครงสร้างของพจนานุกรมที่ได้นำมาใช้ในงานวิจัยนี้คือ โครงสร้างข้อมูลแบบทรัย (Trie) ซึ่งจากการนำโครงสร้างทรัยเข้ามาใช้สามารถช่วยลดขนาดของพจนานุกรมได้ และนอกจากนี้โครงสร้างแบบทรัยนี้ยังสามารถสืบค้นหาคำศัพท์ได้อย่างรวดเร็วและสามารถจะเพิ่มเติมคำศัพท์ได้อย่างสะดวกและรวดเร็ว

สรุปจากงานนี้ได้มีการนำโครงสร้างทรัยมาประยุกต์ใช้เพื่อลดขนาดของพจนานุกรม ซึ่งจากการเปรียบเทียบประสิทธิภาพในด้านความรวดเร็วและขนาดของพจนานุกรม ปรากฏว่าผลการเปรียบเทียบขนาดของพจนานุกรม จำนวน 5400 คำสามารถใช้เนื้อที่ 27,975 ไบต์ ซึ่งมีขนาดน้อยกว่างานวิจัยการแบ่งพยางค์ด้วยพจนานุกรม [4] ซึ่งใช้เนื้อที่ประมาณ 32,482 ไบต์ ส่วนความซับซ้อนของขั้นตอนวิธีในการสืบค้นก็ลดลงด้วยเนื่องมาจากลักษณะทางโครงสร้างของทรัย

- งานวิจัยของสมปรารถนา รัทยานนท์ [7] เป็นการนำพจนานุกรมมาช่วยในการตัดคำภาษาไทย โดยใช้โครงสร้างข้อมูลแบบสองแถวลำดับ (double array) มีขั้นตอนการทำงานดังนี้

1. สร้างแถวลำดับของคำศัพท์ที่ค้นหาจากพจนานุกรมของประโยคอินพุท

2. ดึงคำศัพท์ในแถวลำดับชุดแรกมาแล้วสร้างแถวลำดับของคำศัพท์ที่ค้นหาได้จากพจนานุกรมของประโยคใหม่ (ประโยคใหม่นี้คือการนำคำศัพท์ที่ดึงมาจากแถวลำดับชุดแรกตัดออกจากประโยคอินพุท) กระทำดังนี้เรื่อยไปจนกระทั่งสิ้นสุดประโยคอินพุท และให้แถวลำดับของคำศัพท์ชุดสุดท้ายที่สร้างขึ้นเป็นแถวลำดับที่ n

3. ย้อนการทำงานกลับมาที่แถวลำดับชุดที่ n-1 โดยนำคำศัพท์ในแถวลำดับชุดที่ n-1 ที่ยังไม่ได้ใช้งานมาสร้างแถวลำดับของคำศัพท์ชุดที่ n ใหม่ และใช้คำศัพท์ในแถวลำดับชุดที่ n-1 มาสร้างแถวลำดับของคำศัพท์ชุดใหม่เรื่อยไปจนกระทั่งคำศัพท์ในแถวลำดับชุดที่ n-1 หมด

4. เมื่อคำศัพท์ในแถวลำดับชุดที่ n-1 หมดให้ย้อนการทำงานกลับมาแถวลำดับชุดที่ n-2 แล้วเลือกคำศัพท์ในแถวลำดับชุดที่ n-2 ที่ยังไม่ได้ถูกเลือก ต่อจากนั้นสร้างแถวลำดับคำศัพท์ที่ n-1 ใหม่แล้วเลือกคำศัพท์ในแถวลำดับที่ n-1 เพื่อนำไปสร้างแถวลำดับคำศัพท์ในแถวลำดับ n ใหม่

5. ย้อนการทำงานกลับมาที่แถวลำดับชุดก่อนหน้าเรื่อยไปจนกระทั่งถึงแถวลำดับชุดที่ 1 และคำศัพท์ในแถวลำดับชุดที่ 1 หมด

ตัวอย่างการตัดคำโดยใช้พจนานุกรมที่มีโครงสร้างข้อมูลเป็นแบบสองแถวลำดับแสดงในตารางที่ 2.2

	ประโยค	ขั้นตอนการทำงาน	สิ่งที่ได้
1	การมอบรางวัล	ค้นหาคำศัพท์ในพจนานุกรม	แถวลำดับคำศัพท์ กา, การ
2	การมอบรางวัล	นำคำศัพท์จากแถวลำดับที่ 1มาใช้แล้วตัดคำออกจากประโยค	ประโยคใหม่คือ รมอบรางวัล
3	รมอบรางวัล	ค้นหาคำศัพท์ในพจนานุกรม	แถวลำดับคำศัพท์ รม
4	รมอบรางวัล	ดึงคำศัพท์จากแถวลำดับในข้อ 2 มาใช้แล้วตัดคำออกจากประโยค	ประโยคใหม่คือ อบรางวัล
5	อบรางวัล	ค้นหาคำศัพท์ในพจนานุกรม	แถวลำดับคำศัพท์ อบ
6	อบรางวัล	ดึงคำศัพท์จากแถวลำดับในข้อ 5 มาใช้แล้วตัดคำออกจากประโยค	ประโยคใหม่คือ รางวัล
7	รางวัล	ค้นหาคำศัพท์ในพจนานุกรม	แถวลำดับคำศัพท์ รางวัล
8		ตรวจสอบว่าสิ้นสุดประโยค แล้วแสดงประโยคที่มีการตัดคำ	ประโยคที่มีการตัดคำคือ กา รม อบ รางวัล

ตารางที่ 2.2 ตารางแสดงการตัดคำที่มีการใช้พจนานุกรมที่มีโครงสร้างข้อมูลแบบสองแถวลำดับ

	ประโยค	ขั้นตอนการทำงาน	สิ่งที่ได้
9	การมอบรางวัล	ถอยกลับไปดึงคำศัพท์จากแถวลำดับในข้อ 7 มาปรากฏว่าคำศัพท์หมดแล้วจึงถอยกลับไปดึงในแถวลำดับก่อนหน้า (ข้อ 5, 3) ซึ่งปรากฏว่าคำศัพท์ในแถวลำดับทั้งสองข้อหมดแล้ว จึงถอยกลับไปดึงคำศัพท์จากแถวลำดับในข้อ 1 มาใช้	ประโยคใหม่คือ มอบรางวัล
10	มอบรางวัล	ค้นหาคำศัพท์ในพจนานุกรม	แถวลำดับคำศัพท์ มอบ
11	มอบรางวัล	นำคำศัพท์คำแรกมาใช้	ประโยคใหม่คือ รางวัล
12	รางวัล	ค้นหาคำศัพท์ในพจนานุกรม	แถวลำดับคำศัพท์ รางวัล
13		พบว่าสิ้นสุดคำในประโยค แสดงประโยคที่มีการตัดคำ	ประโยคที่มีการตัดคำคือ การ มอบ รางวัล
14		ถอยไปดึงคำศัพท์ในแถวลำดับที่ 1 ปรากฏว่าคำศัพท์หมด แสดงผลลัพธ์ออกมา	

ตารางที่ 2.2 ตารางแสดงการตัดคำที่มีการใช้พจนานุกรมที่มีโครงสร้างข้อมูลแบบสองแถวลำดับ(ต่อ)

จากตารางที่ 2.2 แสดงการตัดคำโดยประโยคที่นำมาตัดคำคือ การมอบรางวัล ผลลัพธ์ที่ได้ออกมามี 2 ประโยคด้วยกันคือ กา รม อบ รางวัล และ การ มอบ รางวัล จะเห็นได้ว่าในงานวิจัยนี้ได้แสดงชุดคำศัพท์ที่เป็นไปได้ทั้งหมดจากการตัดคำของประโยค แต่ไม่ได้ตัดสินว่าประโยคใดน่าจะเป็นประโยคคำตอบที่ดีที่สุด ทำให้ต้องมีการค้นหาวิธีการตัดสินใจว่าประโยคใดน่าจะเป็นคำตอบที่ดีที่สุดต่อไป

- งานวิจัยของวิรัช ศรเลิศล้ำวานิช [6] ได้มีการพัฒนาการตัดคำที่เรียกว่า การตัดคำให้จำนวนน้อยที่สุด (Maximal Matching) ซึ่งขั้นตอนวิธีนี้ จะสามารถแก้ไขความบกพร่องของการตัดคำแบบเลือกคำยาวที่สุดได้ โดยจุดบกพร่องที่กล่าวนี้คือขั้นตอนวิธีการตัดคำแบบเลือกคำยาวที่สุดจะเลือกคำที่ยาวเกินไปตั้งแต่ครั้งแรก ทำให้ข้อความที่ตามมาเกิดข้อผิดพลาดได้ ตัวอย่างเช่น ประโยค ยาวหลากว่าๆ จะตัดคำได้เป็น ยาว หลาก ว่าๆ โดยที่ถูกต้องควรจะตัดเป็น ยาว หลากว่าๆ

หลักการของการตัดคำให้จำนวนน้อยที่สุดคือ ขั้นตอนแรกคือจะทำการตัดคำที่เป็นไปได้ทุกๆ แบบก่อน แล้วหลังจากนั้นก็ให้ประโยคที่มีจำนวนคำน้อยที่สุด ตัวอย่างเช่น ไปหามเหสี สามารถตัดได้เป็น ไป หาม เห สี กับ ไป หา มเหสี ซึ่งเมื่อพิจารณาจำนวนคำแล้ว วิธีการนี้ก็จะ

เลือกประโยค ไป หา เมหสี ซึ่งเป็นประโยคที่ถูกต้อง สำหรับในกรณีที่ตัดคำแล้วเกิดได้จำนวนคำที่เท่ากันก็ให้นำการตัดคำแบบเลือกคำยาวที่สุดเข้ามาช่วยพิจารณา ตัวอย่างเช่นประโยค ปลานอนตากลม สามารถตัดคำได้ทั้งหมด 2 แบบคือ ปลา นอน ตาก ลม และ ปลา นอน ตา กลม ซึ่งจะมีจำนวนคำเท่ากันทั้ง 2 ประโยค แต่เมื่อใช้การตัดคำแบบเลือกคำยาวที่สุดเข้ามาพิจารณา ประโยคที่ได้คือ ปลา นอน ตาก ลม

สรุปวิธีการนี้จะสามารถช่วยแก้ไขข้อบกพร่องของการตัดคำแบบเลือกคำที่ยาวที่สุดได้ เพราะว่าการเลือกคำที่ยาวที่สุดเมื่อเจอข้อความที่กำกวมก่อน โดยไม่มีการพิจารณาถึงข้อความถัดไป ซึ่งมีลักษณะเหมือนการใช้ขั้นตอนวิธีแบบโลภ (Greedy Algorithm) ที่พิจารณาเฉพาะบริเวณใกล้ๆเท่านั้น แต่วิธีการตัดคำให้จำนวนน้อยที่สุดจะเป็นการใช้ขั้นตอนวิธีแบบโลภโดยพิจารณาข้อความทั้งหมดแทน แต่อย่างไรก็ตามเนื่องจากวิธีการนี้ใช้เฉพาะพจนานุกรมในการตัดคำเท่านั้น ดังนั้นการตัดคำนี้ยังไม่สามารถที่จะตัดคำได้ถูกต้องทั้งหมด แต่ถ้าจะให้ถูกต้องทั้งหมดนั้น จำเป็นจะต้องมีการนำโครงสร้างทางไวยากรณ์ หรือความสัมพันธ์ทางความหมายเข้ามาใช้ประกอบในการพิจารณาด้วย

<u>วิธีการใช้คลังข้อความ</u>

จากการพัฒนาการตัดพยางค์และการตัดคำโดยการใช้กฎหรือพจนานุกรมแล้วยังมีการนำความรู้ที่ได้จากคลังข้อความ เช่นค่าสถิติการใช้คำ และลักษณะไวยากรณ์ภายในคลังข้อความ เป็นต้น งานวิจัยที่เกี่ยวกับการตัดคำโดยใช้คลังข้อความมีดังต่อไปนี้

-งานวิจัยของอัศนีย์ ก่อตระกูลและคณะ [14] ในงานวิจัยนี้จะนำเรื่องสถิติเข้ามาใช้แก้ปัญหาการตัดคำและการกำหนดหน้าที่ของคำหรือประเภทย่อยของคำ โดยมีการนำเรื่องแบบจำลองไตรแกรม เข้ามาช่วยในการแก้ปัญหาการตัดคำ การคำนวณค่าความน่าจะเป็นของประโยคโดยใช้แบบจำลองไตรแกรมสามารถคำนวณได้ดังที่แสดงในสมการที่ 1

$$\begin{aligned} P(W) &= \prod_{i=1}^{n} P(w_{i,n}) \\ &= \prod_{i=1}^{n} P(w_i \mid w_{i-1}, w_{i-2}) \end{aligned} \tag{1}$$

จากสมการที่ 1 คือการคำนวณหาค่าความน่าจะเป็นของแต่ละประโยค โดย W คือประโยคที่ตัดคำแล้ว และประโยค W จะประกอบไปด้วยคำต่างๆ ซึ่ง $W = w_1w_2...w_n$ โดยที่ w_i

คือคำศัพท์ และการคำนวณค่าความน่าจะเป็นของแต่ละประโยคจะมีข้อกำหนดว่า ความน่าจะเป็นของ w_i จะขึ้นอยู่กับ w_{i-1} และ w_{i-2} เท่านั้น แต่เนื่องจากการคำนวณค่าความน่าจะเป็นนั้นจะต้องใช้คลังข้อความขนาดใหญ่มาก โดยคลังข้อความควรจะมากกว่า n^3 คำ โดยที่ n คือจำนวนคำที่เป็นไปได้ทั้งหมด สาเหตุที่วิธีการนี้ต้องใช้คลังข้อความที่มีขนาดมากกว่า n^3 คำ เนื่องจากวิธีนี้จะต้องมีการนำค่าสถิติการเกิดของคำ 3 คำที่ติดกันมาใช้ในการคำนวณ ดังนั้นเพื่อให้มีค่าสถิติของการเกิดคำ 3 คำที่ติดกันทุกๆแบบ อย่างน้อยที่สุดจะต้องใช้ n^3 คำ ซึ่งในความจริงเราไม่สามารถหาคลังข้อความขนาดดังกล่าวได้ ทำให้มีการประมาณสมการที่ 1 เป็นสมการที่ 2 แทน

$$\prod_{i=1}^{n} P(w_i \mid w_{i-2,i-1}) = \prod_{i=1}^{n} (\lambda_1 P(w_i) + \lambda_2 P(w_i \mid w_{i-1}) + \lambda_3 P(w_i \mid w_{i-1}, w_{i-2})) \quad (2)$$

จากสมการที่ 2 นี้จะเป็นการแก้ปัญหาเรื่องจำนวนข้อมูลที่นำมาใช้นั้นไม่เพียงพอ โดยจะมีการนำค่าความน่าจะเป็น ของไบแกรม (Bigram) และยูนิแกรม (Unigram) เข้ามาช่วยในการคำนวณด้วย และค่า $\lambda_1 + \lambda_2 + \lambda_3$ ให้มีค่าเท่ากับ 1

- งานวิจัยของสุรพันธ์ เมฆนาวินและคณะ [16] ได้นำวิธีการทางสถิติเข้ามาช่วยในการแก้ไขปัญหาความกำกวม ซึ่งวิธีการทางสถิติที่นำมาใช้คือการใช้ค่าสถิติที่เกิดจากลำดับของหน้าที่คำหรือประเภทย่อยของคำ หรืออาจกล่าวได้ว่าเป็นการนำเอาส่วนหนึ่งของไวยากรณ์ มาใช้ในการแก้ไขปัญหาความกำกวม

การตัดคำโดยใช้หน้าที่คำแบบจำลองไตรแกรม คือการตัดคำโดยมีการนำเอาค่าสถิติ ซึ่งพิจารณาจากความต่อเนื่องของหน้าที่คำหรือประเภทย่อยของคำ ส่วนวิธีการเลือกแบบการตัดคำที่ดีที่สุดนั้นทำได้โดยหาประโยคที่มีความน่าจะเป็นมากที่สุด โดยการหาความน่าจะเป็นของแต่ละประโยคสามารถคำนวณตามสมการที่ 3

$$P(W_i) = \sum_{T} P(W_i, T_i) = \sum_{T} \prod_{i} P(t_i \mid t_{i-1}, t_{i-2}) \times P(w_i \mid t_i) \quad (3)$$

จากสมการที่ 3 W_i คือประโยคที่ตัดคำแล้ว ซึ่งนำมาจากประโยคที่มีคะแนนที่ดีที่สุด N อันดับแรกจากวิธีการตัดคำให้จำนวนน้อยที่สุด และ $W_i = w_1 w_2 .. w_n$ โดย w_i คือคำที่ตัดได้ ส่วน $T_i = t_1 t_2 .. t_n$ โดย t_i คือหน้าที่คำหรือประเภทย่อยของคำของ w_i และ $P(w_i \mid t_i)$ กับ $P(t_i \mid t_{i-1}, t_{i-2})$ สามารถคำนวณได้จากคลังข้อความ สรุปความหมายจากสมการนี้คือการหาแบบ

การตัดคำที่ดีที่สุด โดยพิจารณาจากผลรวมความน่าจะเป็นของหน้าที่คำหรือประเภทย่อยของคำทุกแบบที่เป็นไปได้ของแต่ละประโยค และมีข้อกำหนดว่าความน่าจะเป็นของการเกิดหน้าที่คำหรือประเภทย่อยของคำที่ตำแหน่งปัจจุบันจะขึ้นอยู่กับหน้าที่คำหรือประเภทย่อยของคำของ 2 คำก่อนหน้าเท่านั้น กล่าวอีกนัยหนึ่งคือวิธีการนี้จะไม่สนใจว่าหน้าที่ของคำหรือประเภทย่อยของคำที่ถูกต้องที่สุดจะเป็นอะไร แต่จะสนใจว่าการตัดคำแบบไหนจะดีที่สุด ทำให้วิธีการนี้เหมาะสมสำหรับงานที่ต้องการทราบขอบเขตคำเพียงอย่างเดียวเท่านั้น

สรุปวิธีการนี้จะสามารถแก้ไขปัญหาความกำกวมได้ดีกว่าวิธีการก่อนๆ ที่ได้กล่าวมาทั้งหมด เนื่องจากมีการพิจารณาถึงหน้าที่ของคำหรือประเภทย่อยของคำเข้ามาประกอบด้วย แต่อย่างไรก็ตามในกรณีที่ข้อความกำกวมมีหน้าที่คำหรือประเภทย่อยของคำเหมือนกัน วิธีการนี้ก็ไม่สามารถที่จะแก้ไขปัญหาได้ดีนัก และข้อจำกัดอีกอย่างหนึ่งก็คือเราจะต้องทำการเก็บค่าสถิติจากคลังข้อความ (Corpus) โดยที่คลังข้อความที่ดีควรจะนำมาจากเอกสารหลายประเภท และจะต้องมีขนาดใหญ่พอสมควร ดังนั้นประสิทธิภาพของวิธีการตัดคำแบบนี้จะขึ้นอยู่กับคลังข้อความด้วย

- งานวิจัยของอัศนีย์ ก่อตระกูลและคณะ [15] ได้ทำการแก้ไขปัญหาเรื่องคำศัพท์ที่ไม่ปรากฏในพจนานุกรม โดยไม่ได้แค่หาขอบเขตของคำเท่านั้น แต่ยังสามารถที่จะบอกถึงหน้าที่คำและแสดงถึงลักษณะทางความหมาย (Semantic Attribute) และยังสามารถที่แก้ไขคำในกรณีที่เกิดการสะกดผิดด้วย ซึ่งในงานวิจัยนี้มีการนำวิธีการทางสถิติ (Statistical Model) และมีการนำกฎต่างๆ เข้ามาช่วยในการพิจารณาด้วย

ขั้นตอนวิธีในการแก้ปัญหาเรื่องคำที่ไม่ปรากฏในพจนานุกรม ประกอบด้วย 3 ขั้นตอนซึ่งแสดงดังต่อไปนี้

1. ทำการตัดคำโดยใช้แบบจำลองไตรแกรม [14] ซึ่งเมื่อทำการตัดคำแล้วผลลัพธ์ที่ได้สามารถแบ่งออกได้เป็น 2 กรณีคือ

1.1 กรณีที่เกิดข้อความที่ไม่ปรากฏในพจนานุกรม

1.2 กรณีที่ไม่เกิดข้อความที่ไม่ปรากฏในพจนานุกรม

2. ถ้าผลลัพธ์จากข้อ 1 ที่ได้เป็นกรณีที่ 1.1 ให้ไปทำขั้นตอนที่ 3 แต่ถ้าเป็นในกรณีที่ 1.2 ให้ใช้ แบบจำลองการแบ่งโดยใช้ความหมาย (Semantic Segmenting Model) ซึ่งสามารถคำนวณได้ดังสมการที่ 4

$$\arg\max_{t_{1,n}} P(w_{1,n}, t_{1,n}) = \prod_{i=1}^{n} P(t_i \mid t_{i-1}, t_{i-2}) * P(w_i \mid t_i) \qquad (4)$$

โดย $w_{1,n}$ จะหมายถึงประโยคที่แบ่งคำแล้วได้ออกมาเป็น w_1 ถึง w_n และ $t_{1,n}$ คือ ลำดับแท็กความหมาย (Semantic tag) โดย t_i คือแท็กความหมายของ w_i ซึ่งในสมการนี้จะทำการหาแท็กความหมายของแต่ละคำ ที่จะทำให้ค่าความน่าจะเป็นของ $P(w_{1,n}, t_{1,n})$ มีค่ามากที่สุด แล้วนำค่ามาเปรียบเทียบกับค่าขีดเริ่มเปลี่ยน (Threshold) ตามเงื่อนไขดังต่อไปนี้

2.1 $P(w_{1,n}, t_{1,n}) \geq$ ค่าขีดเริ่มเปลี่ยน จะหมายความว่าไม่มีการเกิดคำศัพท์ที่ไม่ปรากฏในพจนานุกรมขึ้น

2.2 $P(w_{1,n}, t_{1,n}) <$ ค่าขีดเริ่มเปลี่ยน แล้วให้เลือกคำที่ทำให้ $P(w_{i,i+3}, t_{i,i+3})$ มีค่าน้อยที่สุด และให้ไปทำขั้นตอนที่ 3 ต่อไป

3. ขั้นตอนนี้จะเป็นการหาขอบเขตของคำศัพท์ที่ไม่ปรากฏในพจนานุกรมและบอกถึงหน้าที่คำและความหมายคำ ซึ่งภายในขั้นตอนนี้จะประกอบด้วยขั้นตอนย่อย 4 ขั้นตอนคือ

3.1 การทายขอบเขตโดยใช้วิทยาการศึกษาสำนึก

3.2 สร้างเซตของตัวเลือกของคำศัพท์ที่ไม่ปรากฏในพจนานุกรม โดยใช้กฎที่มีการพิจารณาบริบท (Context Sensitive Rules) และมีการพิจารณาลักษณะของตัวอักษร

3.3 ลองแทนที่ส่วนที่น่าสงสัยว่าจะเป็นคำที่ไม่มีในพจนานุกรม ด้วยตัวเลือกต่างๆ (Unknown Word Candidate) สำหรับวิธีการสร้างตัวเลือกของคำที่ไม่ปรากฏในพจนานุกรมนั้น จะอธิบายในส่วนถัดไป

3.4 คำนวณค่าความน่าจะเป็น โดยใช้สมการที่ 4

ถ้า $P(w_{1,n}, t_{1,n})$ >= ค่าขีดเริ่มเปลี่ยน แสดงว่าคำที่เลือกเป็นคำที่ถูกต้อง แต่ถ้า $P(w_{1,n},t_{1,n}) > P(w_{1,n},t_{1,n})$ ก่อนหน้า ให้กลับไปทำขั้นตอนที่ 3.3 สำหรับในกรณีอื่นๆ แสดงว่ามีข้อผิดพลาดเกิดขึ้น

วิธีการสร้างตัวเลือกของคำศัพท์ที่ไม่ปรากฏในพจนานุกรมสามารถจะสร้างได้ดังต่อไปนี้

1. เมื่อมีการตัดคำแล้วเกิดข้อความที่ไม่ปรากฏในพจนานุกรมจำนวน 2 ชุดที่อยู่ใกล้กัน โดยห่างกันไม่เกิน 2 ตัวอักษร ก็ให้สร้างคำใหม่ โดยรวมข้อความที่ไม่ปรากฏในพจนานุกรมและคำที่อยู่ระหว่างข้อความทั้ง 2 เข้าด้วยกัน

2. เมื่อทำการตัดคำแล้วพบข้อความที่ไม่ปรากฏในพจนานุกรม ก็ให้สร้างคำใหม่ ซึ่งสามารถจะสร้างได้ทั้งหมด 4 แบบคือ

2.1 ให้ข้อความนั้นเป็นคำเลย

2.2 สร้างคำใหม่โดยนำข้อความนั้นมารวมกับคำข้างหน้า

2.3 สร้างคำใหม่โดยนำข้อความนั้นมารวมกับคำถัดไป

2.4 สร้างคำใหม่โดยนำข้อความนั้นมารวมกับคำข้างหน้าและคำถัดไป

- งานวิจัยของไพศาล เจริญพรสวัสดิ์ [3] เป็นงานวิจัยที่ได้ทำการแก้ไขปัญหาความกำกวมและปัญหาคำศัพท์ที่ไม่ปรากฏในพจนานุกรมในกรณีที่เป็นชื่อเฉพาะ ที่ประกอบด้วยชื่อคน ชื่อองค์กร หรือชื่อสถานที่เท่านั้น โดยมีการนำลักษณะทางไวยากรณ์ภายในคลังข้อความคือ ใช้คุณลักษณะของคำบริบท (Context Word) และสิ่งที่เกิดร่วมกันโดยมีลำดับ (Collocation) ไปเรียนรู้ด้วยระบบ RIPPER และ Winnow ซึ่งในการแก้ไขปัญหาความกำกวม ข้อความหรือคำที่นำมาพิจารณาคือข้อความที่กำกวมและคำบริบทสำหรับปัญหานี้คือ คำรอบๆข้อความที่กำกวมภายใน +/-10 คำ สำหรับปัญหาคำศัพท์ที่ไม่ปรากฏในพจนานุกรม ข้อความหรือคำที่จะนำมาพิจารณาคือตัวเลือกของคำที่ไม่ปรากฏในพจนานุกรม (Unknown Word Candidate) ส่วนบริบทที่ใช้สำหรับปัญหานี้คือ คำรอบๆตัวเลือกของคำที่ไม่ปรากฏในพจนานุกรมภายใน +/-10 คำ นอกจากนี้ยังได้แบ่งประเภทคำที่ไม่ปรากฏในพจนานุกรมเป็น 2 ประเภทใหญ่คือ 1. คำศัพท์ที่ไม่ปรากฏในพจนานุกรมอย่างชัดเจน (Explicit Unknown Word) คือภายในคำนั้นๆจะไม่มีข้อความส่วนใดภายในคำนั้นที่เป็นคำที่พบอยู่ในพจนานุกรม ตัวอย่างเช่น โลตัส คาร์ฟูร์ สุนีย์ เป็นต้น และ 2. คำศัพท์ที่ไม่ปรากฏในพจนานุกรมอย่างซ่อนเร้น (Hidden Unknown Word) คือภายในคำนั้นๆจะมีส่วนหนึ่งส่วนใดหรือทุกส่วนเป็นคำที่พบอยู่ในพจนานุกรมเช่น สุมานี สมชาย เป็นต้น

ขั้นตอนวิธีการแก้ไขปัญหาความกำกวม มีขั้นตอนต่อไปนี้

1. นำประโยคมาตัดคำโดยใช้แบบจำลองไตรแกรม
2. เลือกประโยคที่มีคะแนนดีที่สุด N ประโยค
3. นำกฎที่ได้จากริปเปอร์ หรือโครงข่ายวินโนว์มาช่วยในการแก้ปัญหา

ขั้นตอนวิธีการแก้ไขปัญหาคำศัพท์ที่ไม่ปรากฏในพจนานุกรม มีขั้นตอนดังต่อไปนี้

1. นำประโยคมาทำการตัดคำโดยใช้แบบจำลองไตรแกรม
2. เลือกประโยคที่ดีที่สุด N ประโยค
3. ทำการค้นหาบริเวณที่น่าจะเกิดคำศัพท์ที่ไม่ปรากฏในพจนานุกรม ซึ่งจะกล่าวถึงในส่วนถัดไป
4. สร้างตัวเลือกคำศัพท์ที่ไม่ปรากฏในพจนานุกรม ซึ่งก็จะกล่าวถึงในส่วนถัดไป
5. สร้างประโยคใหม่จากประโยคเดิมโดยนำตัวเลือกไปแทนที่
6. กำกับหน้าที่คำโดยแบบจำลองไตรแกรม
7. นำกฎที่ได้จากริปเปอร์หรือโครงข่ายวินโนว์ เข้ามาใช้ในการเลือกตัวเลือกที่มีคะแนนมากที่สุด

วิธีการค้นหาบริเวณที่น่าจะเกิดคำศัพท์ที่ไม่ปรากฏในพจนานุกรม มีวิธีดังนี้

1. หาค่าความน่าจะเป็นของ $P(w_i \mid t_i)$ ที่มีค่าน้อยกว่าค่าขีดเริ่มเปลี่ยน (threshold)
2. หาค่าความน่าจะเป็นของ $P(t_i \mid t_{i-1}, t_{i-2})$ ที่มีค่าน้อยกว่าค่าขีดเริ่มเปลี่ยน

วิธีการสร้างตัวเลือกคำศัพท์ที่ไม่ปรากฏในพจนานุกรมคือ

1. ให้ข้อความนั้นเป็นคำเลย
2. สร้างคำใหม่โดยนำข้อความนั้นมารวมกับคำข้างหน้าตั้งแต่คำที่ 1 ถึงคำที่ 4
3. สร้างคำใหม่โดยนำข้อความนั้นมารวมกับคำถัดไปตั้งแต่คำที่ 1 ถึงคำที่ 4
4. สร้างคำใหม่โดยนำข้อความนั้นมารวมกับคำข้างหน้าและคำถัดไป ตั้งแต่คำที่1 ถึงคำที่ 4

งานวิจัยนี้สรุปว่าการนำคุณลักษณะเข้ามาใช้ในการแก้ไขปัญหาความกำกวมและปัญหาคำศัพท์ที่ไม่ปรากฏในพจนานุกรมมีประสิทธิภาพที่ดีกว่าวิธีการตัดคำแบบจำลองไตรแกรมและการตัดคำให้จำนวนน้อยที่สุด นอกจากนี้การแก้ปัญหาคำศัพท์ที่ไม่ปรากฏในพจนานุกรมอย่างซ่อนเร้นทุกส่วนจะแก้ปัญหาได้ต่ำกว่าแบบอื่นๆ เนื่องจากการค้นหาคำศัพท์ประเภทนี้ทำได้ยากกว่าแบบอื่นๆ ซึ่งจะต้องหาวิธีการค้นหาบริเวณที่เกิดคำศัพท์ประเภทนี้ให้มีประสิทธิภาพที่ดีต่อไป