บทที่ 2

งานวิจัยและทฤษฎีที่เกี่ยวข้อง

เนื่องจากการตัดคำได้มีการพัฒนาติดต่อกันมาเป็นเวลายาวนาน ทำให้มีงานวิจัยด้านการตัดคำเกิดขึ้นมากมายหลายวิธี ซึ่งในช่วงแรกนั้นได้มีการพัฒนาการตัดพยางค์ขึ้นมาก่อน หลังจากนั้นค่อยมีการพัฒนาการตัดคำตามมา ซึ่งในบทนี้จะกล่าวถึงวิธีงานวิจัยด้านการตัดพยางค์ การตัดคำและงานที่เกี่ยวข้องที่ผ่านมาในตั้งแต่อดีตจนถึงปัจจุบัน ในวิทยานิพนธ์นี้จะแบ่งวิวัฒนาการของการตัดคำหรือตัดพยางค์ที่ผ่านมาโดยแบ่งตามลักษณะฐานข้อมูลที่จะนำมาใช้ในการตัดคำ ซึ่งสามารถแบ่งได้เป็น 3 ยุคคือ 1. ยุคการใช้กฎ 2. ยุคการใช้พจนานุกรม 3. ยุคการใช้คลังข้อความ

2.1 ยุคการใช้กฎ

ในยุคนี้คอมพิวเตอร์ยังไม่มีความสามารถในประมวลผลสูงมากนัก ประกอบกับหน่วยความจำในเครื่องคอมพิวเตอร์มีขนาดเล็ก ทำให้ในยุคนี้มีการพัฒนาการตัดพยางค์ขึ้นมาก่อน เนื่องจากพยางค์นั้นมีกฎเกณฑ์ที่แน่นอนมากกว่าคำ ทำให้ในยุคนี้มีการนำกฎเข้ามาใช้ในการตัดพยางค์ ซึ่งจากการนำกฎเข้ามาใช้ในการตัดพยางค์แล้วผลปรากฏว่าจะสามารถแบ่งพยางค์ได้ถูกต้องจำนวนมาก โดยวิธีการแบ่งพยางค์นั้นมีการพัฒนาขึ้นมากมาย โดยงานการแบ่งพยางค์ที่ผ่านมามีดังต่อไปนี้

2.1.1 งานของ ยุพิน ไทยรัตนานนท์

งานของยุพิน ไทยรัตนานนท์ (Yupin Thairatananond, 1981) เป็นงานวิจัยการตัดพยางค์ โดยการใช้กฎในการตัดพยางค์ ซึ่งกฎต่างๆ ที่สร้างขึ้นมานั้นโดยอาศัยหลักไวยากรณ์ภาษาไทย แต่ก็จะมีปัญหาในการสร้างกฎเพราะมีบางพยางค์ไม่เป็นไปตามกฎที่ตั้งไว้ ทำให้มีการจัดเก็บพยางค์ต่างๆ ที่เป็นข้อยกเว้นไว้ในแฟ้มข้อมูล ซึ่งงานวิจัยนี้ได้พัฒนาโดยใช้ภาษาพีแอลวัน (PL/I)

ลักษณะของกฎที่นำมาใช้ในการตัดพยางค์ภายในงานวิจัยนี้ ได้สร้างมาจากลักษณะไวยากรณ์ทางภาษาไทย โดยมีการพิจารณาจากลักษณะของอักขระที่ปรากฏในพยางค์หรือคำ ซึ่งทำให้มีการจัดหมวดหมู่ตัวอักขระภาษาไทย 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. กลุ่มอักขระพิเศษ (Special character)

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

2.1.2 งานของ สุรินทร์ จรรยาพรพงษ์

สุรินทร์ จรรยาพรพงษ์ (Surin Chamyapompong, 1983) ได้ทำการวิจัยเกี่ยวกับการตัดคำภาษาไทยโดยใช้พยางค์ 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)

เนื่องจากลักษณะของตัวอักษรภาษาไทยนั้นสามารถจะเป็นจุดที่บอกขอบเขตของพยางค์ได้อย่างดี ทำให้ในงานวิจัยนี้มีการนำลักษณะของตัวอักษรมาสร้างกฎการตัดพยางค์ซึ่งเรียกกฎเหล่านี้ว่า กฎที่ได้จากคุณสมบัติของอักษรหรือกฎกลุ่มเอ

ตัวอย่างกฎที่ได้จากคุณสมบัติของอักษรในการหาขอบเขตหน้าของพยางค์ เช่น

กฎ A-1F : สระต่างๆ เหล่านี้ ะ า ิ ี ึ ื ุ ู ำ ั และ วรรณยุกต์ ่ ้ ๊ ๋ จะต้องมีพยัญชนะอยู่ข้างหน้าอย่างน้อย 1 ตัวอักษรเสมอ

กฎ A-2F : สระ เ แ ไ โ ส่วนใหญ่จะเป็นตัวอักษรแรกในพยางค์เสมอ ยกเว้นคำบางคำ ตัวอย่างเช่น ขโมย , ขโลม , ทแยง , จเร , สไบ เป็นต้น

กฎ A-3F : สระ ใ จะเป็นตัวอักษรแรกของพยางค์เสมอ โดยไม่มีข้อยกเว้น

ตัวอย่างกฎที่ได้จากคุณสมบัติของอักษรในการหาขอบเขตหลังของพยางค์ เช่น

กฎ A-1T : พยัญชนะต่อไปนี้ ศ ณ ญ ช ฐ ฏ ฏ ฒ ฬ ฒ จะเป็นตัวสะกดเสมอ ยกเว้นพยางค์ดังต่อไปนี้ ศก ศร ศง ศพ ชก ฐก ฬส ญวน ณรงค์ ศตวรรษ และยังมีพยางค์อื่นๆ อีก แต่พยางค์เหล่านั้นสามารถจัดการได้โดยใช้ กฎ A-1F

กฎ A-2T : สระ ็ จะตัวมีตัวสะกดหนึ่งตัวเสมอ ยกเว้น พยางค์เหล่านี้ " วิ พิ อี"

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

จากตัวอย่างที่แสดงข้างต้นจะเห็นว่าการหาขอบเขตของพยางค์สามารถจะนำคุณสมบัติของตัวอักษรต่างๆ มาสร้างเป็นกฎเพื่อใช้ในการแบ่งพยางค์ได้ นอกจากนี้ในงานวิจัยนี้ยังมีการสร้างกฎที่ใช้ในการแบ่งพยางค์ โดยกฎที่ได้จะพิจารณาจากคุณสมบัติของรูปแบบการใช้สระแต่ละตัวหรือกฎกลุ่มบี ซึ่งจะพิจารณาจากลักษณะของพยางค์ในภาษาไทย โดยจากที่สังเกตเห็นคือลักษณะของพยางค์ในภาษาไทยนั้นจะมีรูปแบบที่แน่นอนและตายตัว ทำให้ในงานวิจัยนี้มีการสร้างกฎต่างๆ โดยอ้างอิงจากรูปแบบของพยางค์ต่างๆ ที่ปรากฏในภาษาไทย

ตัวอย่างกฎที่ได้จากคุณสมบัติของรูปแบบการใช้สระ ในการหาขอบเขตหลังของพยางค์ เช่น

กฎ B-1T : สระเหล่านี้ " ั ิ ึ ื " ถ้ามีวรรณยุกต์ แล้วจะต้องมีตัวสะกด 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

นอกจากการสร้างกฎจากคุณสมบัติของตัวอักษรและคุณสมบัติของรูปแบบการใช้สระแล้ว ในงานวิจัยนี้มีการสร้างกฎสำหรับจัดการกับพยางค์ที่ไม่ใช่ลักษณะของพยางค์ภาษาไทย ซึ่งพยางค์เหล่านั้นอาจจะประกอบไปด้วย อักขระพิเศษ ตัวเลข คำย่อ หรือ อักขระจากภาษาต่างชาติ

จากการทดลองวัดความถูกต้องในการตัดพยางค์ งานวิจัยนี้ได้ทดลองกับเอกสารต่างๆ จำนวน 100 เล่ม โดยเอกสารนั้นได้นำมาจากเอกสารชนิดต่างๆ จำนวน 10 ชนิด ดังเช่น หนังสือพิมพ์

ปรัชญา วิทยา-ศาสตร์ ศาสนา ภาษาศาสตร์ ฯลฯ และจากการทดสอบปรากฏว่าสามารถตัดพยางค์ได้ถูกต้องถึง 96%

2.2 ยุคการใช้พจนานุกรม

ในยุคนี้ถือได้ว่าเป็นยุคเริ่มแรกในการตัดคำ เนื่องจากในยุคนี้เครื่องคอมพิวเตอร์ได้มีการพัฒนาขึ้นและมีหน่วยความจำมากขึ้น จากยุคที่แล้วมีการนำเอากฎเข้ามาช่วยในการแบ่งพยางค์ แต่สำหรับการแบ่งคำแล้วการใช้กฎอย่างเดียวไม่สามารถที่จะหาขอบเขตของคำได้ ทำให้ในยุคนี้ได้มีการคิดค้นหาวิธีการแบ่งคำโดยมีการนำเอาพจนานุกรมเข้ามาใช้ร่วมกับกฎในการตัดคำด้วย โดยแนวคิดการตัดคำโดยใช้พจนานุกรมแบบต่างๆ มีดังนี้

2.2.1 ยืน ภู่วรวรรณ และวิวรรธน์ อึ่มอารมณ์

ในงานวิจัยนี้จะเป็นงานวิจัยการแบ่งพยางค์ด้วยพจนานุกรม (ยืน ภู่วรวรรณและ วิวรรธน์ อึ่มอารมณ์, 2529) ซึ่งถือได้ว่าเป็นงานวิจัยงานแรกของการตัดพยางค์ที่มีการนำพจนานุกรมเข้ามาใช้ โดยจะจัดเก็บพยางค์ต่างๆ ไว้ในพจนานุกรม และมีการนำกฎไวยากรณ์ต่างๆ จำนวน 18 กฎเข้ามาช่วยในกรณีที่ไม่พบพยางค์ในพจนานุกรม

หลักการทำงานของกระบวนวิธีการตัดพยางค์ด้วยพจนานุกรมนี้ก็คือ จะทำการตรวจสอบสายอักขระ (String) ที่เข้ามาจากซ้ายไปขวากับพยางค์ที่ได้เก็บไว้ในพจนานุกรม ในกรณีที่ทำการตรวจสอบแล้วปรากฏว่าพบพยางค์มากกว่า 1 พยางค์ในพจนานุกรม ก็ให้ทำการเลือกแบ่งพยางค์โดยเลือกพยางค์ที่ยาวที่สุด แล้วก็ทำต่อไปเรื่อยๆ จนจบสายอักขระ แต่ถ้าในกรณีที่เลือกพยางค์ที่ยาวที่สุดไปแล้ว ทำให้เกิดพยางค์ที่ไม่ปรากฏในพจนานุกรมก็ยอมให้มีการย้อนรอย (Back Tracking) กับไปเลือกพยางค์ที่ยาวรองลงมาแทน ซึ่งวิธีการนี้จะเป็นที่รู้จักกันในชื่อ การตัดคำ(พยางค์)แบบเลือกคำ(พยางค์)ยาวที่สุด (Longest Matching)

จากงานวิจัยนี้ได้มีการเปรียบเทียบความรวดเร็วในการแบ่งพยางค์ ซึ่งสรุปผลได้ว่าเมื่อนำพจนานุกรมเข้ามาใช้ในการแบ่งพยางค์จะสามารถตัดพยางค์ได้รวดเร็วกว่าการใช้กฎ โดยที่ความถูกต้องของการตัดพยางค์นั้นสามารถตัดได้ถูกต้องมากกว่า 99 % แต่สำหรับวิธีการนี้ก็ยังมีข้อเสียคือ ต้องเสียเนื้อที่ในการจัดเก็บพจนานุกรมในหน่วยความจำหลักเป็นจำนวน 50 กิโลไบต์แต่ก็สามารถเก็บข้อมูลพจนานุกรมไว้ในเครื่องคอมพิวเตอร์ในสมัยนั้นได้

2.2.2 งานของ ดวงแก้ว สวามิภักดิ์

งานวิจัยชิ้นนี้คือ การสร้างซอฟต์แวร์วิเคราะห์ไวยากรณ์ไทยภายใต้ระบบยูนิกซ์ (ดวงแก้ว สวามิภักดิ์, 2533) เป็นงานวิจัยด้านการตัดคำภาษาไทย โดยใช้กฎทางไวยากรณ์ที่สร้างขึ้นมาเอง และมีการนำพจนานุกรมเข้ามาใช้ประกอบร่วมด้วย โดยสาเหตุที่นำทั้งกฎไวยากรณ์และพจนานุกรมเข้ามาช่วยในการตัดคำนั้นก็เพื่อที่จะแก้ไขปัญหาการตัดโดยใช้พจนานุกรมเพียงอย่างเดียว (ยืน ภู่วรวรรณและวิวรรธน์ อิ่มอารมณ์, 2529) ซึ่งไม่สามารถตัดคำได้ถูกต้องในกรณีที่คำนั้นไม่มีอยู่ในพจนานุกรม

งานวิจัยการตัดคำนี้ ได้ทำภายใต้ระบบปฏิบัติการยูนิกซ์ และได้มีการนำโปรแกรมเล็กซ์ (Lex) เข้ามาช่วยจัดการการตัดคำ โดยมีการสร้างกฎต่างๆ ให้อยู่ในรูปแบบนิพจน์ที่มีกฎเกณฑ์ (Regular Expression) โดยกฎที่สร้างขึ้นมานี้ประกอบไปด้วย 43 กฎ (รายละเอียดของกฎต่างๆ สามารถดูได้ในภาคผนวก ก) ซึ่งกฎที่ได้มานี้จะไม่มีการรวมตัวสะกดเข้าไปในกฎด้วยยกเว้นบางกรณี เนื่องจากลักษณะของโปรแกรมเล็กซ์ จะพยายามสร้างกลุ่มตัวอักษร (Token) ที่มีขนาดที่ยาวที่สุดก่อน ดังนั้นถ้ามีการนำกฎที่มีตัวสะกดเข้ามาใช้ จะเป็นสาเหตุให้มีการรวมเอาอักษรตัวหน้าของคำถัดไปมาเป็นตัวสะกดได้ ซึ่งเมื่อได้ผ่านการวิเคราะห์ด้วยกฎแล้ว ขั้นตอนต่อไปก็จะมีการรวมกลุ่มตัวอักษรเข้าด้วยกัน โดยทำการตรวจสอบจากพจนานุกรม ส่วนโครงสร้างของพจนานุกรมที่นำมาใช้ในที่นี้คือฐานข้อมูลแบบรีเลชั่น (Relational DBMS) ซึ่งใช้คำเป็นดรรชนี (Index) และไฟล์ดรรชนีได้พัฒนาขึ้นโดยใช้โครงสร้างข้อมูลแบบบีทรี (B-Tree)

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

2.2.3 สัมพันธ์ ระรื่นรมย์

ในงานวิจัยนี้เป็นงานวิจัยการแบ่งคำไทยด้วยพจนานุกรม (สัมพันธ์ ระรื่นรมย์, 2534) โดยเป้าหมายของงานวิจัยนี้จะเน้นที่การเพิ่มประสิทธิภาพในด้านความรวดเร็วของขั้นตอนวิธีในการตัดคำและการลดขนาดของพจนานุกรม เนื่องจากเมื่อนำพจนานุกรมเข้ามาใช้ในการตัดคำแล้วจะทำให้ความถูกต้องในการตัดคำเพิ่มขึ้นมากกว่าการตัดคำใช้กฎอย่างเดียว ดังนั้นในงานวิจัยนี้จึงไม่ได้เน้นการเพิ่มประสิทธิภาพในด้านความถูกต้องมากนักเพราะถือว่าการตัดคำโดยใช้พจนานุกรมจะให้ค่าความถูกต้องที่สูงอยู่แล้ว

โดยรายละเอียดของขั้นตอนวิธีการตัดคำนั้นจะมีวิธีการคล้ายกับงานวิจัยการแปงพยางค์โดยใช้ดิกชันนารี (ยืน ภู่วรวรรณและวิวรรธน์ อิ่มอารมณ์, 2529) ซึ่งในงานวิจัยนี้จะทำการจัดเก็บคำลงในพจนานุกรมแทนพยางค์ ส่วนขั้นตอนวิธีจะทำงานเหมือนเดิม คือใช้ขั้นตอนวิธีแบบเลือกคำที่ยาวที่สุดดังที่ได้กล่าวไปแล้ว ตัวอย่างการตัดคำโดยเลือกคำที่ยาวที่สุดแสดงดังตารางที่ 2-1

ตารางที่ 2-1 ตารางแสดงการตัดคำแบบเลือกคำที่ยาวที่สุด

ประโยค	คำที่ได้	คำที่ถูกเลือก
โคนมนอนบนกองหญ้า	โค, โคน	โคน
มนอนบนกองหญ้า	-	(ย้อนรอย)
โคนมนอนบนกองหญ้า	โค, โคน	โค (เลือกคำรองลงมา)
นมนอนบนกองหญ้า	นม	นม
นอนบนกองหญ้า	นอ, นอน	นอน
บนกองหญ้า	บน	บน
กองหญ้า	กอ, กอง	กอง
หญ้า	หญ้า	หญ้า

จากตารางที่ 2-1 จะแสดงการตัดคำแบบเลือกคำที่ยาวที่สุด โดยประโยคที่นำมาตัดคำคือ "โคนมนอนบนหญ้า" สามารถตัดคำได้เป็น "โค นม นอน บน หญ้า"

ส่วนโครงสร้างของพจนานุกรมที่ได้นำมาใช้ในงานวิจัยนี้คือ โครงสร้างข้อมูลแบบทรัย (Trie) ซึ่งจากการนำโครงสร้างทรัยเข้ามาใช้สามารถช่วยลดขนาดของพจนานุกรมได้ และนอกจากนี้โครงสร้างแบบทรัยนี้ยังสามารถสืบค้นหาคำศัพท์ได้อย่างรวดเร็วและสามารถจะเพิ่มเติมคำศัพท์ได้อย่างสะดวกและรวดเร็วด้วย โดยในรายละเอียดต่างๆ เกี่ยวกับโครงสร้างแบบทรัยจะอธิบายเพิ่มเติมในบทที่ 4 เรื่องโครงสร้างของพจนานุกรม

สรุปจากงานนี้ได้มีการนำโครงสร้างทรัยมาประยุกต์ใช้เพื่อลดขนาดของพจนานุกรม ซึ่งจากการเปรียบเทียบประสิทธิภาพในด้านความรวดเร็วและขนาดของพจนานุกรม ปรากฏว่าผลการเปรียบเทียบขนาดของพจนานุกรม จำนวน 5400 คำสามารถใช้เนื้อที่ 27975 ไบต์ ซึ่งมีขนาดน้อยกว่างานวิจัยการแปงพยางค์ด้วยพจนานุกรม (ยืน ภู่วรวรรณและวิวรรธน์ อึ่มอารมณ์, 2529) ซึ่งใช้เนื้อที่ประมาณ 32,482 ไบต์ ส่วนความซับซ้อนของขั้นตอนวิธีในการสืบค้นก็ลดลงด้วยเนื่องมาจากลักษณะทางโครงสร้างของทรัย

2.2.4 วิรัช ศรเลิศล้ำวานิช

สำหรับในงานวิจัยการตัดคำภาษาไทยชิ้นนี้ ได้มีการพัฒนาการตัดคำโดยเรียกว่า "การตัดคำโดยเลือกแบบเหมือนมากที่สุด (Maximal Matching)" (วิรัช ศรเลิศล้ำวานิช, 2536) ซึ่งขั้นตอนวิธีนี้ จะ

สามารถแก้ไขความบกพร่องของการตัดคำแบบเลือกคำยาวที่สุดได้ โดยจุดบกพร่องที่กล่าวนี้คือขั้นตอนวิธีการตัดคำแบบเลือกคำยาวที่สุดจะเลือกคำที่ยาวเกินไปตั้งแต่ครั้งแรก ทำให้ข้อความที่ตามมาเกิดข้อผิดพลาดได้ ตัวอย่างเช่น ประโยค "ไปหามเหลี" จะตัดคำได้เป็น "ไป หาม เห ลี" โดยที่ถูกต้องควรจะตัดเป็น "ไป หา มเหลี"

หลักการของการตัดคำโดยเลือกแบบเหมือนมากที่สุดคือ ขั้นตอนแรกคือจะทำการตัดคำที่เป็นไปได้ทุกๆ แบบก่อน แล้วหลังจากนั้นก่อนให้ประโยคที่มีจำนวนคำน้อยที่สุด ตัวอย่างเช่น "ไปหามเหลี" สามารถตัดได้เป็น "ไป หาม เห ลี" กับ "ไป หา มเหลี" ซึ่งเมื่อพิจารณาจะจำนวนคำแล้ว วิธีการนี้ก็จะเลือกประโยค "ไป หา เมหลี" ซึ่งเป็นประโยคที่ถูกต้อง สำหรับในกรณีที่ตัดคำแล้วเกิดได้จำนวนคำที่เท่ากันก็ให้นำการตัดคำแบบเลือกคำยาวที่สุดเข้ามาช่วยพิจารณา ตัวอย่างเช่นประโยค "ฉันนั่งตากลม" สามารถตัดคำได้ทั้งหมด 2 แบบคือ "ฉัน นั่ง ตาก ลม" และ "ฉัน นั่ง ตา กลม" ซึ่งจะมีจำนวนคำเท่ากันทั้ง 2 ประโยค แต่เมื่อใช้การตัดคำแบบเลือกคำยาวที่สุดเข้ามาพิจารณา ประโยคที่ได้คือ "ฉัน นั่ง ตาก ลม"

สรุปวิธีการนี้จะสามารถช่วยแก้ไขข้อบกพร่องของการตัดคำแบบเลือกคำที่ยาวที่สุดได้ เพราะว่าการเลือกคำที่ยาวที่สุดเมื่อเจอข้อความที่กำกวมก่อน โดยไม่มีการพิจารณาถึงข้อความถัดไป ซึ่งมีลักษณะเหมือนการใช้ขั้นตอนวิธีแบบโลภ (Greedy Algorithm) ที่พิจารณาเฉพาะบริเวณใกล้ๆเท่านั้น แต่วิธีการตัดคำโดยเลือกแบบเหมือนมากสุดจะเป็นการใช้ขั้นตอนวิธีแบบโลภโดยพิจารณาข้อความทั้งหมดแทน แต่อย่างไรก็ตามเนื่องจากวิธีการนี้ใช้เฉพาะพจนานุกรมในการตัดคำเท่านั้น ดังนั้นการตัดคำนี้ยังไม่สามารถที่จะตัดคำได้ถูกต้องทั้งหมด แต่ถ้าจะให้ถูกต้องทั้งหมดนั้น จำเป็นจะต้องมีการนำโครงสร้างทางไวยากรณ์หรือความสัมพันธ์ทางความหมายเข้ามาใช้ประกอบในการพิจารณาด้วย

2.3 ยุคการใช้คลังข้อความ

จากการพัฒนาการตัดคำในยุคที่ผ่านมาเราใช้เพียงกฎ หรือพจนานุกรมในการแบ่งคำเท่านั้น ทำให้การตัดคำในยุคก่อนไม่สามารถที่จะตัดคำได้ถูกต้องทั้งหมด และในยุคนี้ (ปัจจุบัน) เครื่องคอมพิวเตอร์มีประสิทธิภาพมากยิ่งขึ้น มีหน่วยความจำมากขึ้นเป็นจำนวนมาก และได้มีการพัฒนาคลังข้อความ (Corpus) ขึ้นจำนวนมาก ทำให้ในยุคนี้ได้มีการพัฒนาการตัดคำขึ้นมาใหม่ โดยนอกเหนือการใช้กฎ พจนานุกรมแล้วยังมีการนำความรู้ต่างๆ จากคลังข้อความเข้ามาประยุกต์ใช้ด้วย ตัวอย่างความรู้ที่ได้จากคลังข้อความเช่น ค่าสถิติการใช้คำภายในคลังข้อความและลักษณะไวยากรณ์ที่ใช้ในคลังข้อความ เป็นต้น การตัดคำโดยใช้คลังข้อความในยุคนี้มีการพัฒนาในแบบต่างๆ ดังต่อไปนี้คือ

2.3.1 อัศนีย์ ก่อตระกูลและคณะ

ในงานนี้ทำการวิจัยเรื่อง "A Statistical Approach to Thai Word Filtering" (Asanee Kawtrakul et al. ,1997) เนื่องจากปัญหาของการวิเคราะห์หน่วยคำ (Morphological Analysis) สำหรับภาษาไทยนั้นจะมีปัญหาต่างๆ ดังนี้คือ ปัญหาความกำกวม ปัญหาความกำกวมของการกำหนดหน้าที่คำ (Part-of-Speech Tagging Ambiguity) และปัญหาการสะกดคำผิด โดยปัญหาต่างๆ เหล่านี้จะทำให้เกิดผลลัพธ์ที่ตัดคำและกำหนดหน้าที่ของคำแล้วออกมาหลายๆ รูปแบบ ซึ่งในงานวิจัยนี้จะทำการลดผลลัพธ์ที่ไม่เหมาะสมออกไป เพื่อที่สามารถจะทำให้พาสเซอร์(Parser) ทำงานได้รวดเร็วขึ้น

ในงานวิจัยนี้จะนำเรื่องสถิติเข้ามาใช้แก้ปัญหาการตัดคำและการกำหนดหน้าที่ของคำ โดยมีการนำเรื่องโมเดลไตรแกรม เข้ามาช่วยในการแก้ปัญหาการตัดคำ การคำนวณค่าความน่าจะเป็นของประโยคโดยใช้โมเดลไตรแกรมสามารถคำนวณได้ดังที่แสดงในสมการที่ 2-1

$$P(W) = \prod_{i=1}^{n} P(w_{i,n}) = \prod_{i=1}^{n} P(w_i \mid w_{i-1}, w_{i-2}) \quad (2\text{-}1)$$

จากสมการที่ 2-1 คือการคำนวณหาค่าความน่าจะเป็นของแต่ละประโยค โดย W คือประโยคที่ตัดคำแล้ว และประโยค W จะประกอบไปด้วยคำต่างๆ ซึ่ง $W = w_1 w_2 ... w_n$ โดยที่ w_i คือคำศัพท์ และการคำนวณค่าความน่าจะเป็นของแต่ละประโยคจะมีข้อกำหนดว่า ความน่าจะเป็นของ w_i จะขึ้นอยู่กับ w_{i-1} และ w_{i-2} เท่านั้น

แต่เนื่องจากการคำนวณค่าความน่าจะเป็นตามสมการ 2-1 นั้นจะต้องใช้คลังข้อความขนาดใหญ่มาก โดยคลังข้อความควรจะมากกว่า n^3 คำ โดยที่ n คือจำนวนคำที่เป็นไปได้ทั้งหมด สาเหตุที่วิธีการนี้ต้องใช้คลังข้อความที่มีขนาดมากกว่า n^3 คำ เนื่องจากวิธีนี้จะต้องมีการนำค่าสถิติการเกิดของคำ 3 คำที่ติดกันมาใช้ในการคำนวณ ดังนั้นเพื่อให้มีค่าสถิติของการเกิดคำ 3 คำที่ติดกันทุกๆแบบ อย่างน้อยที่สุดจะต้องใช้ n^3 คำ ซึ่งในความจริงเราไม่สามารถหาคลังข้อความขนาดดังกล่าวได้ ทำให้มีการประมาณสมการที่ 2-1 เป็นสมการที่ 2-2 แทน

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

จากสมการที่ 2-2 นี้จะเป็นการแก้ปัญหาเรื่องจำนวนข้อมูลที่นำมาใช้นั้นไม่เพียงพอ โดยจะมีการนำค่าความน่าจะเป็น ของไบแกรม (Bigram) และยูนิแกรม (Unigram) เข้ามาช่วยในการคำนวณด้วย และค่า λ_1 , λ_2 ,λ_3 ให้มีค่าเท่ากับ 0.1, 0.3, 0.6 ตามลำดับ ซึ่งได้นำมาจาก (Charniak, 1996)

สรุปผลจากงานวิจัยนี้สามารถลดรูปแบบของการตัดคำที่ไม่เหมาะสมลงไปได้จำนวนมาก ส่งผลให้งานการวิเคราะห์หน่วยคำนั้นสามารถจะทำงานได้รวดเร็วขึ้น

2.3.2 สุรพันธ์ เมฆนาวินและคณะ

จากการตัดคำที่ผ่านมาจะมีการนำเอาพจนานุกรมเข้ามาใช้ในการตัดคำเพียงอย่างเดียวเท่านั้น และการนำวิทยาการศึกษาลำนึก (Heuristics) ต่างๆ เข้ามาช่วยแก้ปัญหาความกำกวมที่เกิดขึ้นนั้นไม่สามารถที่จะแก้ปัญหาความกำกวมได้ทั้งหมด ดังนั้นจึงได้มีการพัฒนาการตัดคำขึ้น โดยมีการนำวิธีการทางสถิติเข้ามาช่วยในการแก้ไขปัญหาคำกำกวม ซึ่งวิธีการทางสถิติที่นำมาใช้คือการใช้ค่าสถิติที่เกิดจากลำดับของหน้าที่คำ หรืออาจกล่าวได้ว่าเป็นการนำเอาส่วนหนึ่งของไวยากรณ์ มาใช้ในการแก้ไขปัญหาความกำกวม

การตัดคำโดยใช้หน้าที่คำแบบไตรแกรมโมเดล (Surapant Meknavin, Paisarn Charoenpornsawat and Boonserm Kijsirikul, 1997) คือการตัดคำโดยมีการนำเอาค่าสถิติ ซึ่งพิจารณาจากความต่อเนื่องของหน้าที่คำ ส่วนวิธีการเลือกแบบการตัดคำที่ดีที่สุดนั้นทำได้โดยหาประโยคที่มีความน่าจะเป็นมากที่สุด โดยการหาความน่าจะเป็นของแต่ละประโยคสามารถคำนวณตามสมการที่ 2-3

$$\begin{aligned} P(W_l) &= \sum_{T} P(W_l, T_l) \\ &= \sum_{T} \prod_{i} P(t_i \mid t_{i-1}, t_{i-2}) \times P(w_i \mid t_i) \end{aligned} \qquad (2\text{-}3)$$

จากสมการที่ 2-1 W_i คือประโยคที่ตัดคำแล้ว ซึ่งนำมาจากประโยคที่มีคะแนนที่ดีที่สุด N อันดับแรก โดยวิธีการตัดคำคือการตัดคำโดยเลือกแบบเหมือนมากที่สุด 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 คือหน้าที่คำของ w_i และ $P(w_i \mid t_i)$ กับ $P(t_i \mid t_{i-1}, t_{i-2})$ สามารถคำนวณได้จากคลังข้อความ สรุปความหมายจากสมการนี้คือการหาแบบการตัดคำที่ดีที่สุด โดยพิจารณาจากผลรวมความน่าจะเป็นของหน้าที่คำทุกแบบที่เป็นไปได้ของแต่ละประโยค และมีข้อกำหนดว่าความน่าจะเป็นของการเกิดหน้าที่คำที่ตำแหน่งปัจจุบันจะขึ้นอยู่กับหน้าที่คำของ 2 คำก่อนหน้าเท่านั้น กล่าวอีกนัยหนึ่งคือวิธีการนี้จะไม่สนใจว่าหน้าที่ของคำที่ถูกต้องที่สุดจะเป็นอะไร แต่จะสนใจว่าการตัดคำแบบไหนจะดีที่สุด ทำให้วิธีการนี้เหมาะสมสำหรับงานที่ต้องการทราบขอบเขตคำเพียงอย่างเดียวเท่านั้น

สรุปวิธีการนี้จะสามารถแก้ไขปัญหาความกำกวมได้ดีกว่าวิธีการก่อนๆ ที่ได้กล่าวมาทั้งหมด เนื่องจากมีการพิจารณาถึงหน้าที่ของคำเข้ามาประกอบด้วย แต่อย่างไรก็ตามในกรณีที่ข้อความกำกวมมีหน้าที่คำเหมือนกัน วิธีการนี้ก็ไม่สามารถที่จะแก้ไขปัญหาได้ดีนัก และข้อจำกัดอีกอย่างหนึ่งก็คือเราจะต้องทำการเก็บค่าสถิติจากคลังข้อความ (Corpus) โดยที่คลังข้อความที่ดีควรจะนำมาจากเอกสารหลายประเภท และจะต้องมีขนาดใหญ่พอสมควร ดังนั้นประสิทธิภาพของวิธีการตัดคำแบบนี้จะขึ้นอยู่กับคลังข้อความด้วย

2.3.3 อัศนีย์ ก่อตระกูลและคณะ

ในยุคนี้เนื่องจากคอมพิวเตอร์มีความรวดเร็วในการประมวลผลมากขึ้น และมีหน่วยความจำขนาดใหญ่ทำให้ ปัญหาที่สำคัญของยุคนี้ไม่ใช่เรื่องความรวดเร็ว หรือการประหยัดเนื้อที่หน่วยความจำ แต่ปัญหาที่สำคัญในยุคนี้คือความถูกต้องของการตัดคำ เพราะในระบบต่างๆ มีความต้องการที่จะได้การตัดคำที่มีประสิทธิภาพสูงที่สุด ซึ่งในยุคนี้ได้มีการพัฒนาตัดคำวิธีการต่างๆ ดังที่ได้กล่าวไปแล้ว แต่วิธีการดังกล่าวก็ยังไม่สามารถที่จัดการกับคำศัพท์ที่ไม่ปรากฏในพจนานุกรมได้ ดังนั้นในยุคนี้จึงได้มีคิดค้นวิธีการที่จะนำมาใช้ในการแก้ปัญหาคำศัพท์ที่ไม่ปรากฏในพจนานุกรมขึ้น โดย อัศนีย์ ก่อตระกูลและคณะ (Asanee Kawtrakul et. al.,1997)

ในงานวิจัยได้ทำแก้ปัญหาเรื่องคำศัพท์ที่ไม่ปรากฏในพจนานุกรม โดยไม่ได้แค่หาขอบเขตของคำเท่านั้น แต่ยังสามารถที่จะบอกถึงหน้าที่คำและแสดงถึงลักษณะทางความหมาย (Semantic Attribute) และยังสามารถที่แก้ไขคำในกรณีที่เกิดการสะกดผิดด้วย ซึ่งในงานวิจัยนี้มีการนำวิธีการทางสถิติ (Statistical Model) และมีการนำกฎต่างๆ เข้ามาช่วยในการพิจารณาด้วย

ขั้นตอนวิธีในการแก้ปัญหาเรื่องคำที่ไม่ปรากฏในพจนานุกรม ประกอบด้วย 3 ขั้นตอนซึ่งแสดงดังต่อไปนี้

2.3.3.1 ทำการตัดคำอย่างง่ายๆ โดยใช้โมเดลไตรแกรม (Asanee Kawtrakul et al., 1995) ซึ่งเมื่อทำการตัดคำแล้วผลลัพธ์ที่ได้สามารถแบ่งออกได้เป็น 2 กรณีคือ

2.3.3.1.1 กรณีที่เกิดข้อความที่ไม่ปรากฏในพจนานุกรม

2.3.3.1.2 กรณีที่ไม่เกิดข้อความที่ไม่ปรากฏในพจนานุกรม

2.3.3.2 ถ้าผลลัพธ์จากข้อ 2.3.3.1 ที่ได้เป็นกรณีที่ 2.3.3.1.1 ให้ไปทำขั้นตอนที่ 2.3.3.3 แต่ถ้าเป็นในกรณีที่ 2.3.3.1.2 ให้ใช้ โมเดลการแบ่งโดยใช้ความหมาย (Semantic Segmenting Model) ซึ่งสามารถคำนวณได้ดังสมการที่ 2-4

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

โดย $w_{1,n}$ จะหมายถึงประโยคที่แปงคำแล้วได้ออกมาเป็น 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

1. ขั้นตอนนี้จะเป็นการหาขอบเขตของคำศัพท์ที่ไม่ปรากฏในพจนานุกรมและบอกถึงหน้าที่คำและความหมายคำ ซึ่งภายในขั้นตอนนี้จะประกอบด้วยขั้นตอนย่อย 4 ขั้นตอนคือ

1.1 การหาขอบเขตโดยใช้วิทยาการศึกษาสำนึก

1.2 สร้างเขตของตัวเลือกของคำศัพท์ที่ไม่ปรากฏในพจนานุกรม โดยใช้กฎที่มีการพิจารณาจากบริบท (Context Sensitive Rules) และมีการพิจารณาลักษณะของตัวอักษร

1.3 ลองแทนที่ส่วนที่น่าสงสัยว่าจะเป็นคำที่ไม่มีในพจนานุกรม ด้วยตัวเลือกต่างๆ (Unknown Word Candidate) สำหรับวิธีการสร้างตัวเลือกของคำที่ไม่ปรากฏในพจนานุกรมนั้น จะอธิบายในส่วนถัดไป

1.4 คำนวณค่าความน่าจะเป็น โดยใช้สมการที่ 2-4

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