บทที่ 5
ขั้นตอนวิธีการแทรกอักขระแบ่งคำ

ในบทที่ 3 ที่ผ่านมาเป็นการกล่าวถึงหน้าที่หลักที่มอดูลการตัดคำควรกระทำ คือ การระบุจุดตัดคำแต่เพียงอย่างเดียว ซึ่งทำให้วิธีการเชื่อมต่อระหว่างมอดูลการตัดคำและโปรแกรมที่นำผลของการตัดคำไปใช้ มีรูปแบบที่เรียบง่ายและมีประโยชน์ใช้สอยได้ในงานต่างๆเป็นวงกว้าง แต่เพื่อให้การใช้มอดูลการตัดคำในงานบางประเภททำได้โดยสะดวก จึงจะเพิ่มวิธีการแทรกอักขระเพื่อแสดงจุดแบ่งคำ ซึ่งโปรแกรมประยุกต์สามารถเลือกที่จะไม่ใช้วิธีการแทรกอักขระแบ่งคำนี้ หากไม่มีความจำเป็น หรืออาจออกแบบวิธีการแทรกอักขระแบ่งคำขึ้นเองเพื่อใช้งานเฉพาะกิจ ทั้งนี้โดยจะไม่มีผลกระทบต่อการใช้มอดูลการตัดคำผ่านตัวเชื่อมโยงที่ได้กำหนดไว้ในบทที่ 3

การที่อักขระแบ่งคำอยู่ปะปนกับข้อมูลจริงและตัวอักขระแบ่งคำเองอาจจะเป็นตัวอักษรเดียวกับข้อมูลได้ การแทรกอักขระแบ่งคำจึงต้องมีขั้นตอนวิธีที่เหมาะสมเพื่อให้สามารถตรวจรู้ความแตกต่างของอักขระแบ่งคำและข้อมูลได้ และเนื่องจากตัวอักขระแบ่งคำเอง สามารถถูกเลือกใช้ได้ตามความต้องการ ขั้นตอนวิธีที่ใช้นี้จึงต้องสามารถใช้ได้กับ ทุกๆตัวอักษรที่เป็นอักขระแบ่งคำด้วย

5.1 การสอดไส้บิต (bit stuffing)

ปัญหาในลักษณะนี้ได้เคยเกิดขึ้นในการทำการสื่อสารข้อมูล (data communication) มาแล้ว กล่าวคือ ในการติดต่อสื่อสารข้อมูลต้องมีการรับส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์แต่ละเครื่อง ซึ่งการรับส่งข้อมูลนี้ จะต้องมีโพรโทคอล (protocol) ในการติดต่อเพื่อควบคุมความผิดพลาดของข้อมูล ข้อมูลที่ส่งผ่านสายการสื่อสาร (communication line) จึงมีทั้งส่วนที่เป็นข้อมูลจริงกับส่วนที่เป็นตัวควบคุมของโพรโทคอลส่งมาด้วยกัน

การแยกข้อมูลจริงออกจากตัวควบคุมของโพรโทคอลจึงเป็นสิ่งจำเป็นเพื่อให้ได้ข้อมูลที่ถูกต้อง ตัวอย่างเช่น ในโพรโทคอล HDLC (high level data link control) และโพรโทรคอล SDLC (synchronous data link control) (Ahuja 1985) ใช้ข้อมูลที่เป็นบิต 1 เรียงติดกัน 6 ตัว เป็นแฟล็กบอกการเริ่มต้นของรหัสควบคุมที่ใช้นำหน้า และปิดท้ายข้อมูลในแต่ละเฟรม (frame) และยังใช้การส่งแฟล็กอย่างต่อเนื่อง เป็นการบอกให้ทางฝ่ายรับรู้ว่ายังมีการติดต่อกันอยู่อีกด้วย

ถ้าหากในข้อมูลจริงมีข้อมูลที่เป็นบิต 1 ติดกัน 6 ตัวอยู่ด้วย ก็อาจจะทำให้เกิด

การเข้าใจผิดพลาดทางฝ่ายรับได้ การสอดไส้บิตเป็นเทคนิคที่ใช้ในการแก้ปัญหานี้ โดยมีหลักการดังนี้ คือ ฝ่ายส่งจะตรวจสอบข้อมูลจริงที่จะส่งออกไปว่า ถ้ามีข้อมูลที่เป็นบิต 1 ติดกันถึง 5 ตัว เมื่อไรก็จะทำการสอดไส้บิต 0 ลงไป 1 ตัว เพื่อป้องกันไม่ให้มีข้อมูลที่มีบิต 1 ติดกัน 6 ตัว ซึ่งจะซ้ำกับข้อมูลที่เป็นตัวควบคุม

ทางฝ่ายรับเมื่อรับข้อมูลก็จะทำการตรวจสอบข้อมูลที่รับเข้ามา ถ้าเป็นข้อมูลที่มีบิต 1 ติดกัน 5 ตัว ก็หมายความว่าบิตที่ตามหลังมาเป็นบิตที่ถูกสอดไส้เข้ามา ก็จะตัดบิตนั้นออก ข้อมูลที่ผ่านการตัดบิตสอดไส้ออกแล้วก็จะเหมือนข้อมูลเดิมทุกประการ

5.2 การฆ่าความหมาย

หลักการฆ่าความหมายนี้เป็นอีกวิธีหนึ่งที่ใช้ในการแยกข้อมูลส่วนที่เป็นข้อมูลจริงกับส่วนที่เป็นตัวควบคุม ตัวอย่างเช่น ในภาษาซีใช้เครื่องหมายหารกลับหลัง (backslash) '\' เป็นตัวควบคุม (Kernighan & Ritchie 77) หมายถึงให้ตัวอักษรที่ตามหลังมีความหมายพิเศษ เช่น \r หมายถึงปัดแคร่ (carriage return) เป็นต้น

ในกรณีที่ต้องการหมายถึงข้อมูลที่เป็นเครื่องหมายหารกลับหลังจริงๆ โดยไม่ต้องการให้มีความหมายพิเศษอื่น ภาษาซีกำหนดให้ต้องทำการฆ่าความหมายของหารกลับหลังก่อน โดยการใส่เครื่องหมายหารกลับหลังอีกตัวหนึ่งติดกัน ก็จะหมายถึงเครื่องหมายหารกลับหลังที่เป็นข้อมูลเพียงตัวเดียว ตัวอย่างเช่น \\r จะหมายถึง \ กับตัวอักษร r ไม่ได้หมายถึงปัดแคร่

5.3 การออกแบบและพัฒนาขั้นตอนวิธีการแทรกอักขระแบ่งคำ

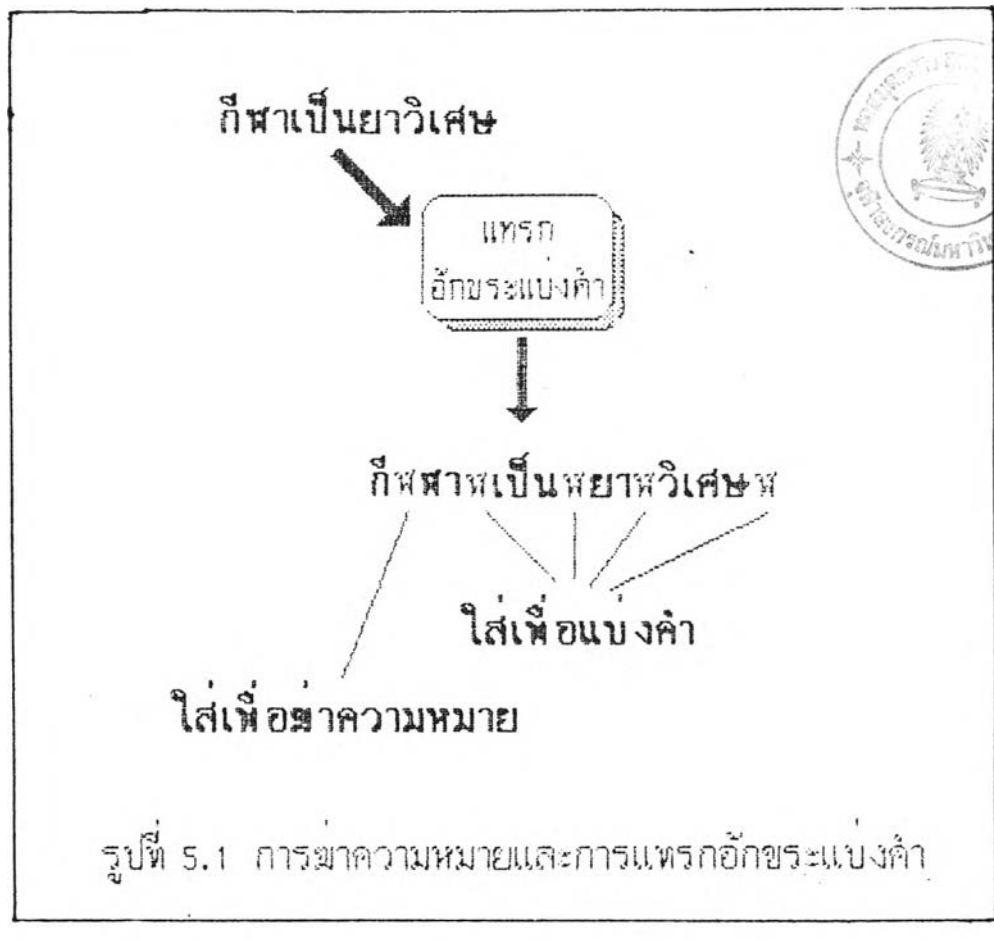
จากหลักการของการสอดไส้บิตและการฆ่าความหมายสามารถนำมาประยุกต์ใช้กับขั้นตอนวิธีการแทรกอักขระแบ่งคำได้

ขั้นตอนวิธีพื้นฐาน

เราสามารถใช้หลักการของการฆ่าความหมายกับขั้นตอนวิธีการแทรกอักขระแบ่งคำ โดยกำหนดให้ทุกๆตัวของข้อมูลที่เหมือนกับอักขระแบ่งคำ ให้ใส่ตัวอักษรนั้นซ้ำลงไปอีกตัวหนึ่ง เนื่องเป็นการฆ่าความหมายของข้อมูลนั้น ไม่ให้หมายถึงตัวอักขระแบ่งคำ ตัวอย่างเช่น ประโยค "กีฬาเป็นยาวิเศษ" สามารถแบ่งคำได้เป็น "กีฬา เป็น ยา วิเศษ" เราจะใส่อักขระแบ่งคำ

ลงหลังคำแต่ละคำ ซึ่งอักขระแบ่งคำในที่นี้ให้เป็นตัวอักษร 'ฯ'

ในข้อความนี้มีข้อมูลที่ซ้ำกับอักขระแบ่งคำอยู่คือคำว่า 'กีฬา' มีตัว 'ฬ' อยู่ จึงต้องทำการฆ่าความหมายของ 'ฬ' ใน 'กีฬา' ก่อน เพื่อไม่ให้สับสนกับตัวอักขระแบ่งคำ ผลลัพธ์ที่ได้จะเป็นดังในรูปที่ 5.1



รูปที่ 5.1 การฆ่าความหมายและการแทรกอักขระแบ่งคำ

จากข้างต้นเราสามารถตั้งเป็นกฎสำหรับการแทรกอักขระแบ่งคำได้ดังนี้

"ให้แทรกอักขระแบ่งคำหลังตัวอักษรสุดท้ายของแต่ละคำและทุกๆตัวอักษรของข้อมูลที่เหมือนกับอักขระแบ่งคำ"

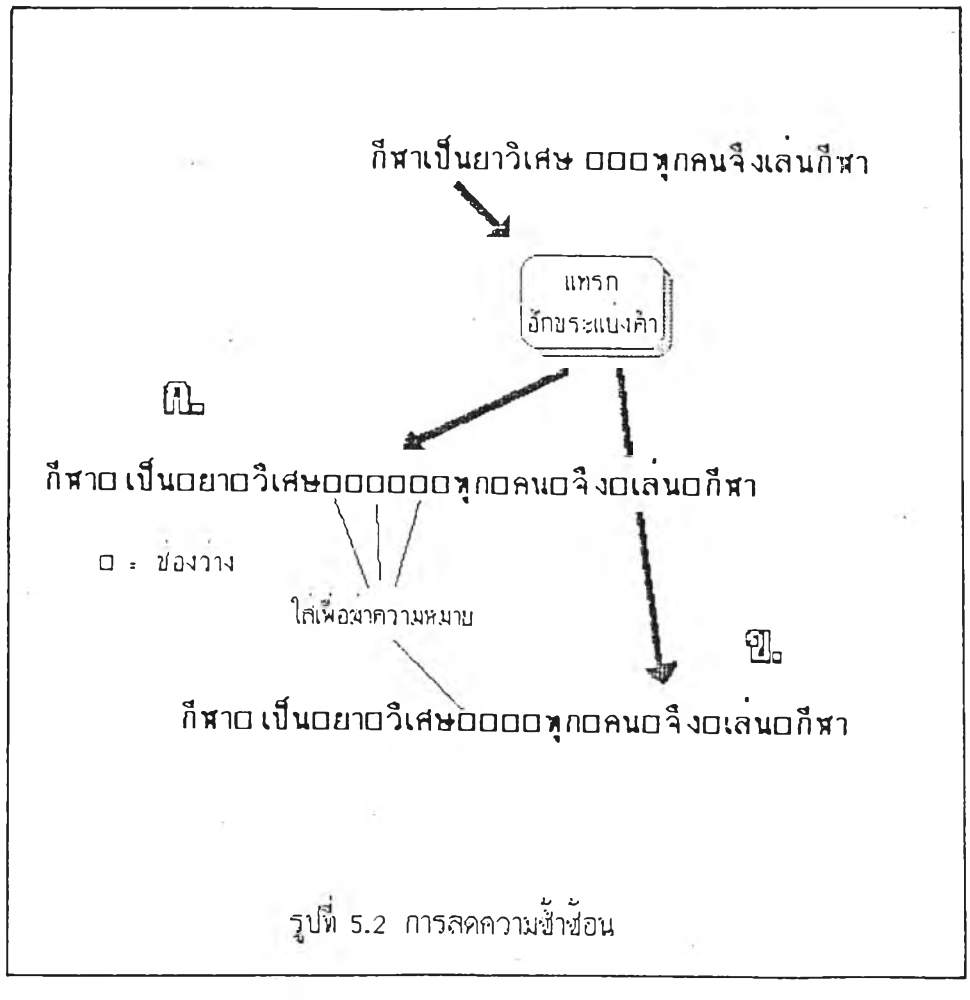
และกฎสำหรับการตรวจสอบว่าส่วนใดเป็นอักขระแบ่งคำและส่วนใดเป็นข้อมูลจริง ก็คือ

"ตัวอักขระแบ่งคำที่อยู่ตัวเดียวโดดๆไม่ติดกับตัวอักขระแบ่งคำอื่น จะหมายถึงตัวอักขระแบ่งคำจริง ส่วนตัวอักขระแบ่งคำที่อยู่ติดกัน 1 คู่จะหมายถึงข้อมูลที่เหมือนตัวอักขระแบ่งคำ 1 ตัว"

กฎข้างต้นเป็นเพียงกฎอย่างกว้างๆ ยังมีข้อบกพร่องอยู่มากดังจะกล่าวต่อไป

<u>การลดความซ้ำซ้อน</u>

อักขระแบ่งคำที่แทรกลงไปเพื่อฆ่าความหมายของข้อมูลที่เหมือนอักขระแบ่งคำนี้ ต้องแทรกลงไปทุกครั้งที่พบตัวอักษรนั้น ในกรณีที่เราเลือกตัวอักขระแบ่งคำที่เป็นข้อมูลที่มีความถี่สูงมาก เช่น เว้นวรรค (space) จะทำให้จำนวนข้อมูลที่ต้องฆ่าความหมายมีปริมาณมาก ข้อความที่ผ่านการแทรกอักขระแบ่งคำแล้วก็จะมีขนาดใหญ่ขึ้นมากด้วย ดังแสดงในรูปที่ 5.2 ก



รูปที่ 5.2 การลดความซ้ำซ้อน

การฆ่าความหมายนั้น เพื่อต้องการไม่ให้มีข้อมูลที่เป็นตัวอักขระแบ่งคำอยู่ตัวเดียวโดดๆ มิฉะนั้นจะทำให้สับสนกับตัวอักขระแบ่งคำจริงได้ ดังนั้นสำหรับข้อมูลที่เหมือนกับอักขระแบ่งคำที่อยู่ติดกันมากกว่า 1 ตัวขึ้นไป เราจึงเพียงแต่เพิ่มอักขระแบ่งคำลงไปฆ่าความหมายเพียง

ตัวเดียวเท่านั้น ไม่จำเป็นต้องฆ่าความหมายของทุกตัวอักษรซึ่งผลลัพธ์ของวิธีนี้จะเป็นดังรูป 5.2 ข

กฎสำหรับการแทรกอักขระแบ่งคำจึงปรับปรุงได้เป็น

"ให้แทรกอักขระแบ่งคำหลังตัวอักษรสุดท้ายของแต่ละคำ และแทรกหน้าทุกกลุ่มของข้อมูลที่เหมือนอักขระแบ่งคำ โดยคำว่ากลุ่ม หมายถึงข้อมูลที่เหมือนอักขระแบ่งคำ ตั้งแต่ 1 ตัวขึ้นไป"

และกฎของการแยกอักขระแบ่งคำกับข้อมูล ก็คือ

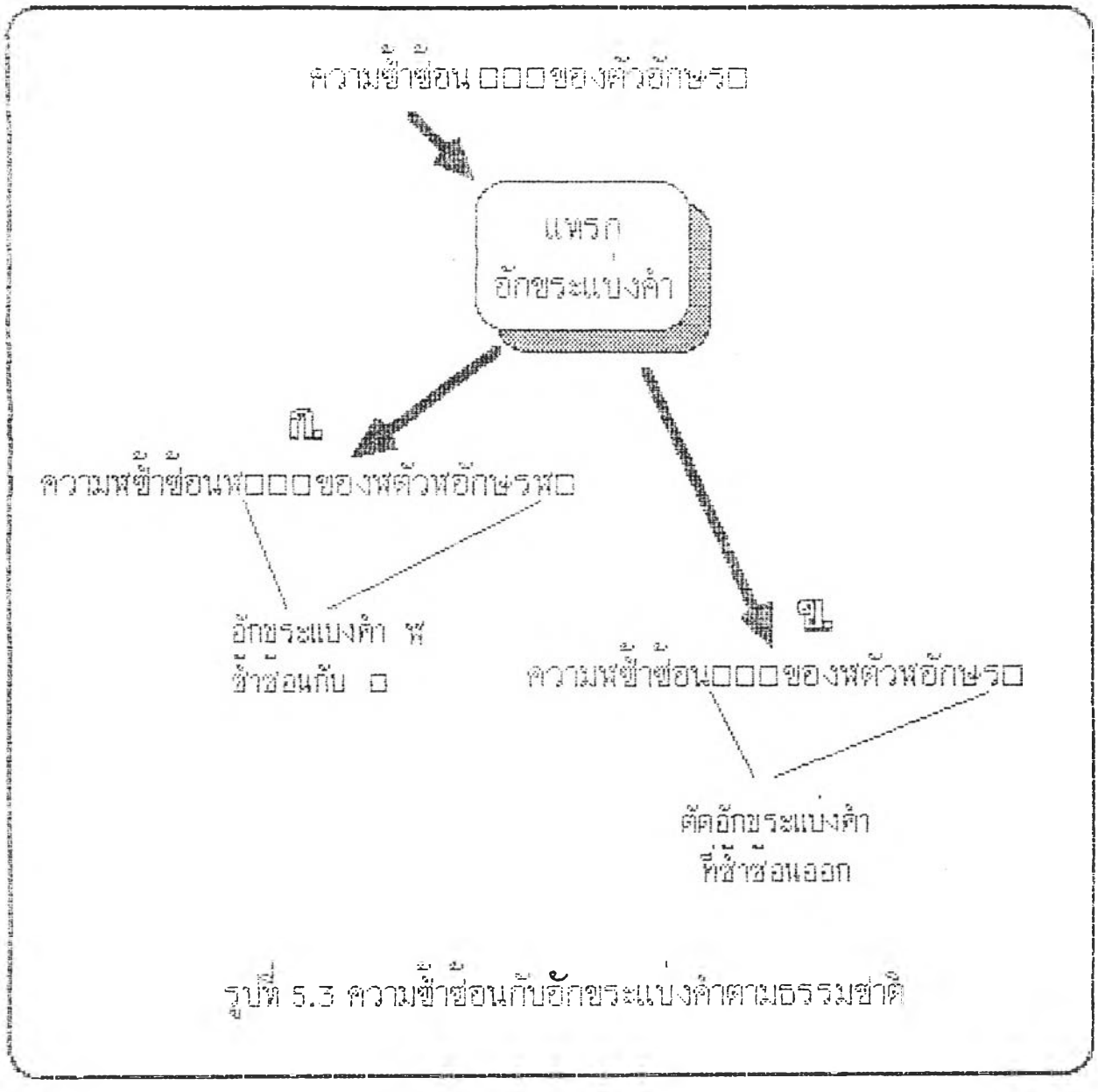
"ตัวอักขระแบ่งคำที่อยู่ตัวเดียวโดดๆ จะเป็นอักขระแบ่งคำจริง ส่วนอักขระแบ่งคำที่อยู่ติดกันตั้งแต่ 2 ตัวขึ้นไป ให้ตัดทิ้งไป 1 ตัว ที่เหลือจะเป็นข้อมูลจริง"

วิธีการที่ใช้ในการลดความซ้ำซ้อนนี้ เป็นการประยุกต์มาจากวิธีการสอดไส้บิตนั่นเอง

นอกจากนี้เรายังสามารถลดความซ้ำซ้อนของการแทรกอักขระแบ่งคำลงได้อีก 1 ชุด นั่นคือ เมื่อจุดสิ้นสุดของคำ อยู่ติดกับข้อมูลที่เป็นตัวอักขระแบ่งคำตามธรรมชาติอยู่แล้ว เช่น เว้นวรรค เราไม่จำเป็นต้องแทรกอักขระแบ่งคำลงไปอีก เพราะจะซ้ำซ้อนกับอักขระแบ่งคำที่มีอยู่แล้ว ดังในรูปที่ 5.3 ก และ ข จะแสดงข้อแตกต่างของทั้ง 2 วิธี

อักขระแบ่งคำพิเศษ (special seperator)

ขั้นตอนวิธีที่พัฒนาขึ้นมาดังข้างต้น สามารถใช้แทรกอักขระแบ่งคำและแยกความแตกต่างของอักขระแบ่งคำกับข้อมูลได้ แต่ยังคงมีจุดอ่อนอยู่คือ กรณีที่ตัวอักขระแบ่งคำเป็นข้อมูลตัวสุดท้ายของคำหรือเป็นตัวแรกของคำ จะทำให้เกิดความสับสนระหว่างตัวอักขระแบ่งคำที่แทรกลงไปหลังตัวสุดท้ายของคำ และตัวอักขระที่แทรกลงไปเพื่อฆ่าความหมายของข้อมูล ตัวอย่างของกรณีนี้เป็นดังรูปที่ 5.4 ในที่นี้ให้อักขระแบ่งคำเป็น 'ง' ในรูปที่ 5.4 ก จะมีข้อมูล 'ง' ที่เป็นตัวสุดท้ายของคำอยู่ ตามขั้นตอนวิธีจะต้องมีการแทรกอักขระแบ่งคำเพื่อฆ่าความหมายของ 'ง' 1 ตัว และเพื่อใส่หลังตัวสุดท้ายของคำ 1 ตัว จึงมีอักขระแบ่งคำอยู่ติดกัน 3 ตัว ทำให้เราไม่สามารถหาอักขระแบ่งคำจริงได้

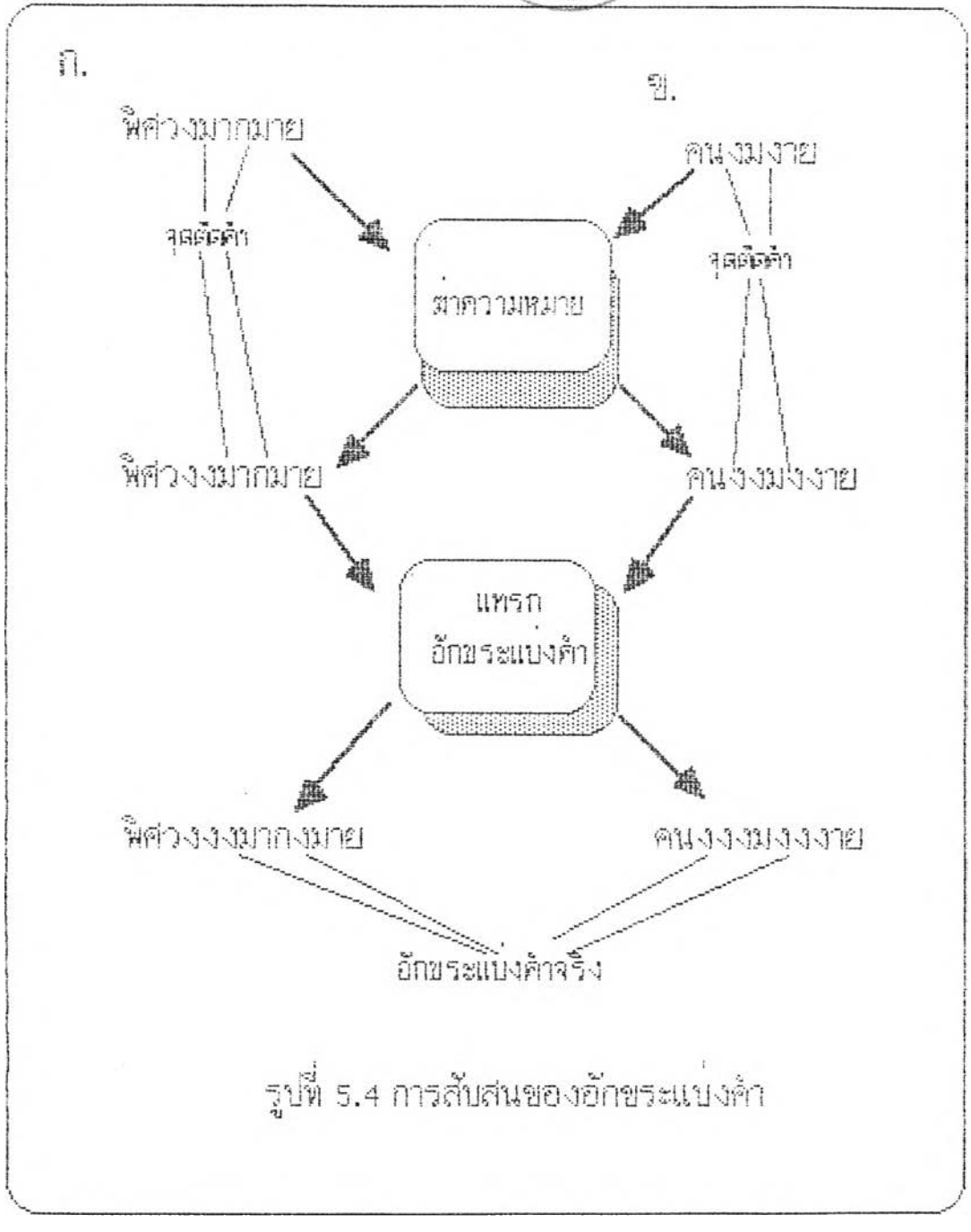


รูปที่ 5.3 ความซ้ำซ้อนกับอักขระแบ่งคำตามธรรมชาติ

สำหรับรูปที่ 5.4 ข ก็เช่นเดียวกัน มีคำที่มี 'ง' เป็นตัวอักษรตัวแรกอยู่ถึง 2 คำ จะได้อักขระแบ่งคำชุดละ 3 ตัว 2 ชุด ซึ่งก็ไม่สามารถแบ่งแยกได้เช่นเดียวกัน

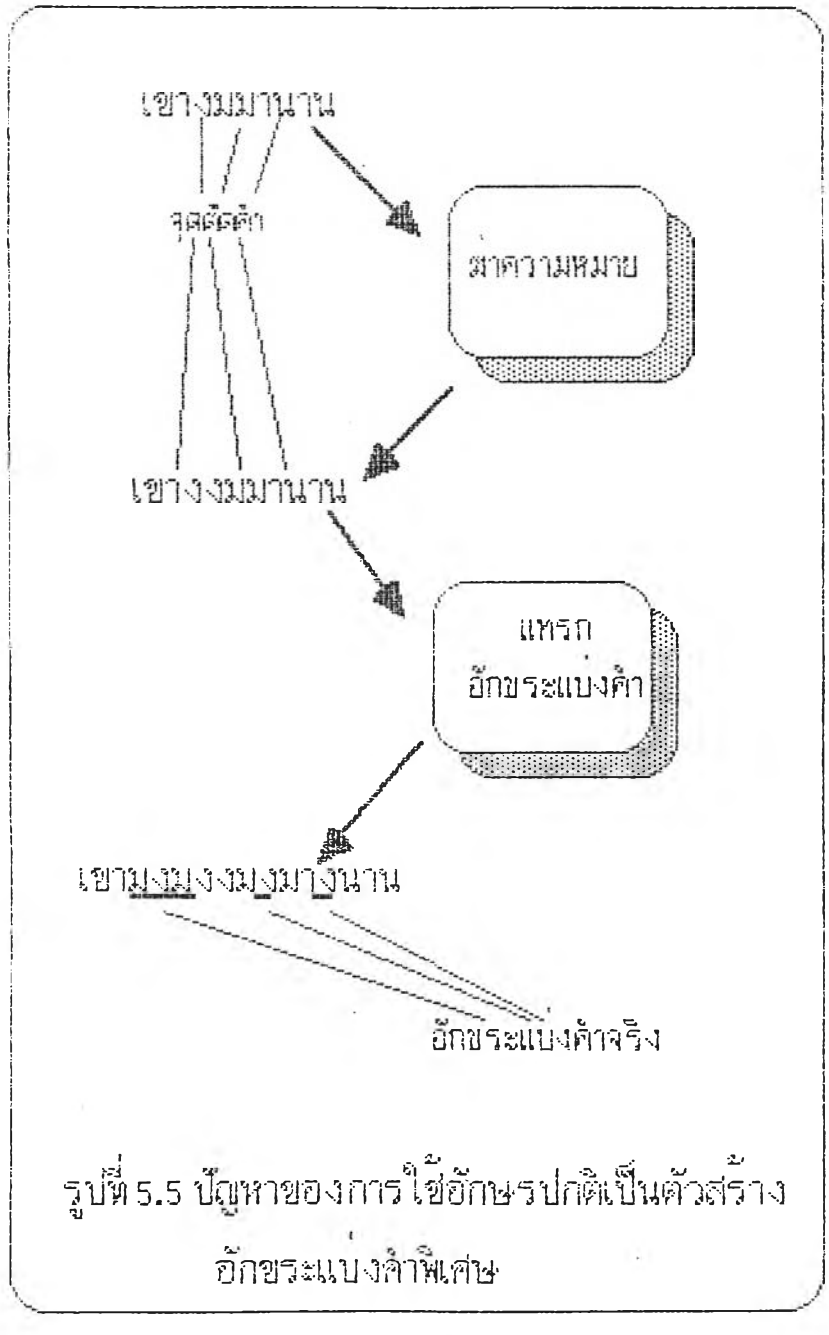
สำหรับในกรณีเหล่านี้ จึงต้องมีอักขระแบ่งคำพิเศษที่แตกต่างไปจากปกติ เพื่อแยกความแตกต่างระหว่างอักขระแบ่งคำ กับอักขระสำหรับฆ่าความหมายและข้อมูลจริง ตัวอักขระแบ่งคำพิเศษนี้จะเป็นแบบ 3 ตัวอักษร โดยใช้ตัวอักขระแบ่งคำเดิมแต่เพิ่มตัวอักษรอื่นลงไป ทั้งด้านหน้าและด้านหลังเพื่อป้องกันการปะปนกับกลุ่มของอักขระแบ่งคำทั้งด้านหน้าและด้านหลัง

ตัวอักษรที่ใช้ประกบหน้าและหลังตัวอักขระแบ่งคำเดิมเพื่อสร้างอักขระแบ่งคำพิเศษนี้ จะต้องเป็นตัวอักษรที่ไม่มีโอกาสเป็นตัวแรกและตัวสุดท้ายของคำ กล่าวคือ ไม่มีโอกาสพบอยู่



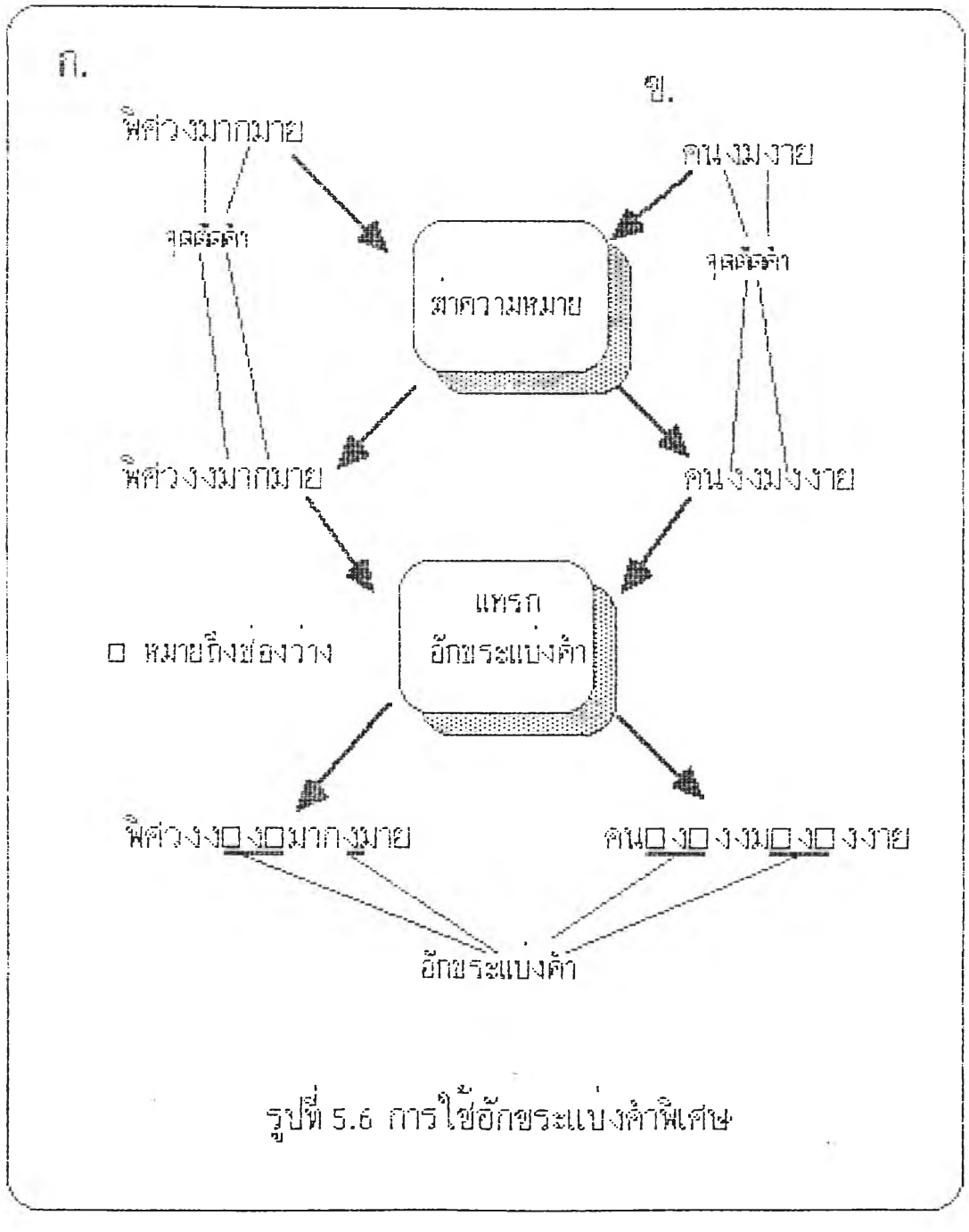
รูปที่ 5.4 การสับสนของอักขระแบ่งคำ

หน้าหรือหลังตัวอักขระแบ่งคำจริงในแบบปกติเพื่อป้องกันความสับสน ดังตัวอย่างในรูปที่ 5.5 ถ้าอักขระแบ่งคำเป็น 'ง' และเราใช้ 'ม' เป็นตัวประกบเพื่อสร้างอักขระแบ่งคำพิเศษ ผลลัพธ์จะออกมาดังรูป ทำให้เราไม่สามารถบอกได้ว่า ชุดของ 'มงม' เป็นอักขระแบ่งคำพิเศษหรือเป็นข้อมูลจริงอยู่ปนกับอักขระแบ่งคำปกติ

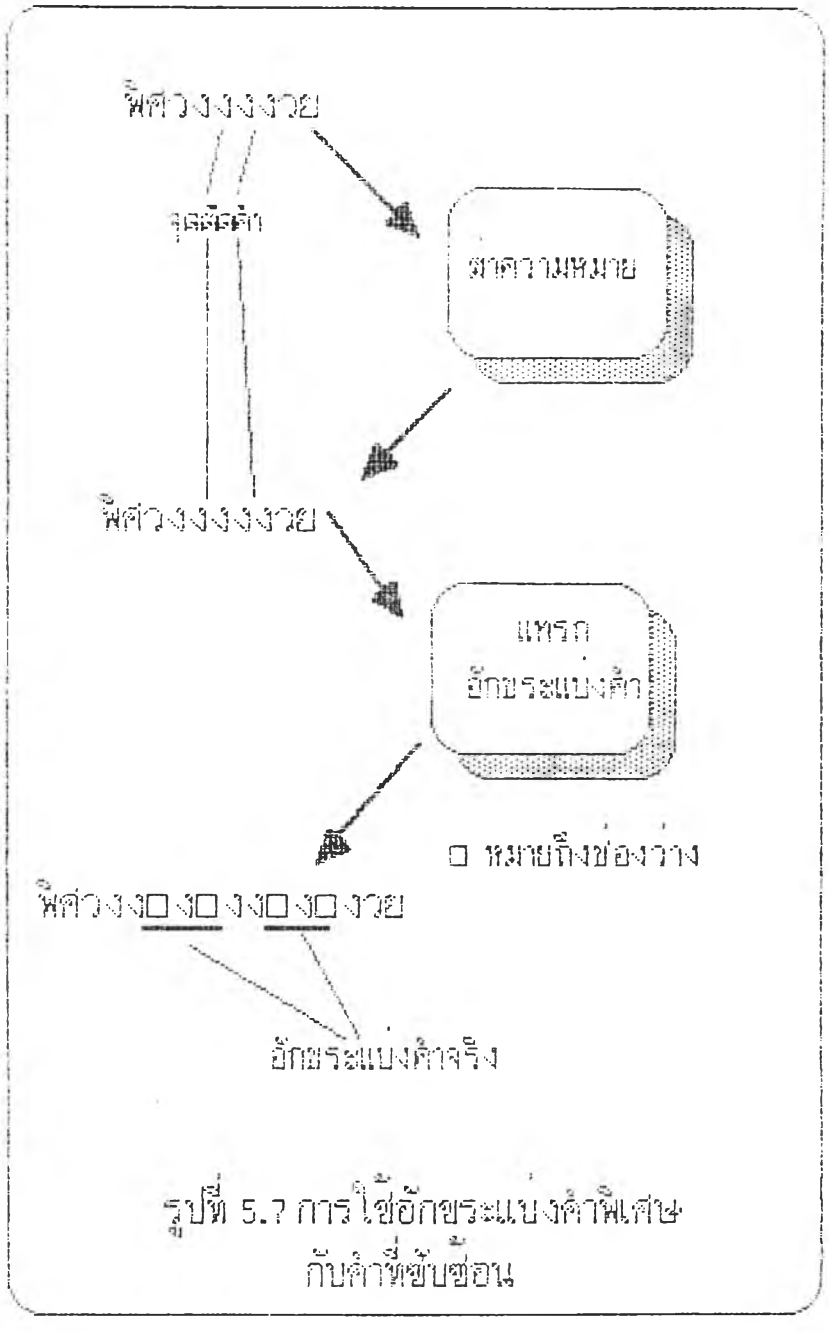


รูปที่ 5.5 ปัญหาของการใช้อักษรปกติเป็นตัวสร้างอักขระแบ่งคำพิเศษ

ตัวอักษรที่ใช้ประกบได้ผล ก็คือตัวอักษรที่เป็นตัวอักขระแบ่งคำตามธรรมชาติ เนื่องจากไม่มีโอกาสเป็นตัวอักษรตัวแรกหรือตัวอักษรตัวสุดท้าย สำหรับขั้นตอนวิธีที่ใช้ในวิทยานิพนธ์ฉบับนี้จะเลือกใช้ตัวเว้นวรรค ดังนั้นปัญหาจากรูปที่ 5.4 ก็จะแก้ไขได้โดยใช้อักขระแบ่งคำชนิดพิเศษ ดังแสดงในรูปที่ 5.6 ก และ ข ส่วนในรูป 5.7 จะแสดงการใช้อักขระแบ่งคำพิเศษในประโยคที่มีคำซับซ้อนมากยิ่งขึ้น



รูปที่ 5.6 การใช้อักขระแบ่งคำพิเศษ



รูปที่ 5.7 การใช้อักขระแบ่งคำพิเศษ
กับคำที่ซับซ้อน

5.4 สรุป

จากการออกแบบและพัฒนาขั้นตอนวิธีการแทรกอักขระแบ่งคำข้างต้นสามารถสรุปได้ดังนี้คือ

กฎสำหรับการแทรกอักขระแบ่งคำ

1. ให้ฆ่าความหมายของข้อมูลที่เหมือนอักขระแบ่งคำ โดยแทรกอักขระแบ่งคำ 1 ตัว ลงหน้าแต่ละชุดของข้อมูลที่เหมือนอักขระแบ่งคำ โดยคำว่าชุดของข้อมูลหมายถึง ข้อมูลตั้งแต่ 1 ตัวขึ้นไป

2. แทรกอักขระแบ่งคำลงหลังตัวอักษรตัวสุดท้ายของแต่ละคำ โดยพิจารณาดังนี้

2.1 ถ้าตัวอักษรสุดท้ายของคำอยู่ติดกับอักขระแบ่งคำตามธรรมชาติ ไม่ต้องแทรกอักขระแบ่งคำ

2.2 ถ้าตัวอักษรสุดท้ายของคำคือข้อมูลที่เหมือนกับอักขระแบ่งคำ หรือตัวอักษรตัวหน้าของคำถัดไปคือข้อมูลที่เหมือนกับอักขระแบ่งคำ ให้แทรกอักขระแบ่งคำพิเศษ (เว้นวรรค, อักขระแบ่งคำ, เว้นวรรค) แทนอักขระแบ่งคำปกติ

2.3 นอกจาก 2 ข้อแรก ให้แทรกอักขระแบ่งคำปกติ

<u>กฎสำหรับการตรวจรู้อักขระแบ่งคำและข้อมูล</u>

1. ตรวจรู้อักขระแบ่งคำจริงโดยใช้กฎเกณฑ์ ดังนี้

1.1. อักขระแบ่งคำพิเศษเป็นอักขระแบ่งคำจริงเสมอ

1.2. อักขระแบ่งคำที่ไม่อยู่ติดกับอักขระแบ่งคำอื่น เป็นอักขระแบ่งคำจริง ยกเว้นกรณีที่ตามหลังอักขระแบ่งคำพิเศษ

2. หลังจากแยกอักขระแบ่งคำจริงออกไปแล้ว ตัดข้อมูลที่เหมือนอักขระแบ่งคำออก 1 ตัว ที่ทุกๆชุดของข้อมูลที่เหมือนอักขระแบ่งคำ ที่เหลือคือข้อมูลจริง