บทที่ 3

การกำกับหน้าที่คำ

การกำกับหน้าที่คำคือการระบุหน้าที่คำของคำที่กำหนดมา ส่วนหน้าที่คำ (Part of Speech: POS) คือสิ่งที่ระบุว่าคำนั้นทำหน้าที่ทางไวยากรณ์เป็นอะไรภายในประโยคหนึ่งๆ โดยคำหนึ่งคำอาจจะมีหลายหน้าที่ได้ขึ้นอยู่กับตำแหน่งภายในประโยคนั้นๆ เช่นคำว่า "ฉัน" สามารถจะมีหน้าที่ได้ 2 อย่างคือ 1. เป็นคำกริยา 2. เป็นคำสรรพนาม ตัวอย่างเช่น "พระฉันเพลก่อนเที่ยง" คำว่า "ฉัน" ในที่นี้ก็จะทำหน้าที่เป็นคำกริยา แต่ถ้าในประโยค "ฉันกับน้องชอบไปดูหนังด้วยกัน" คำว่า "ฉัน" ในที่นี้จะทำหน้าที่เป็นคำสรรพนาม เป็นต้น

สำหรับการวิเคราะห์ทางด้านภาษา ชุดหน้าที่คำ (POS Tag Set) ที่นำมาใช้จะมีผลต่อการวิเคราะห์เป็นอย่างมาก และในความจริงการระบุหน้าที่คำเพียงบอกว่าเป็น คำนาม คำกริยา คำสรรพนาม คำคุณศัพท์ คำวิเศษณ์ ฯลฯ นั้นไม่เพียงพอที่จะนำมาใช้ในการวิเคราะห์ทางภาษาศาสตร์ ดังนั้นนักภาษาศาสตร์จึงได้มีการสร้างชุดหน้าที่คำที่จะมีประสิทธิภาพเพียงพอต่อการนำไปใช้ในการวิเคราะห์ทางภาษาศาสตร์ โดยลักษณะชุดหน้าที่คำของแต่ละภาษานั้นจะมีลักษณะแตกต่างกันไปตามภาษานั้นๆ และในภาษาหนึ่งๆ อาจจะมีชุดหน้าที่คำได้หลายชุดโดยขึ้นอยู่กับแนวคิดของการนำชุดหน้าที่คำไปใช้ ตัวอย่างเช่นในภาษาอังกฤษได้มีการสร้างชุดหน้าที่คำออกมาหลายชุดเช่น ชุดหน้าที่คำเพนทรีแบงค์ (Penn Treebank tagset) ซึ่งในเพนทรีแบงค์นั้นได้แบ่งหมวดหมู่หน้าที่คำออกเป็น 36 ชนิด (Allen, 1995) และ ชุดหน้าที่คำบราวน์ (Brown tagset) ได้แบ่งหมวดหมู่คำออกเป็น 80 ชนิด สำหรับภาษาไทยได้มีสร้างชุดหน้าที่คำออกมาหลายชุดเช่นกัน ตัวอย่างเช่น ชุดหน้าที่คำออร์คิด (Orchid tagset) ซึ่งแบ่งหมวดหมู่คำเป็น 47 ชนิด (Virach Sornlertlamvanich, Thatsanee Charoenporn and Isahara, 1997) และชุดหน้าที่คำของมหาวิทยาลัยเกษตรศาสตร์ เป็นต้น

จากที่ได้กล่าวในบทนำว่า งานด้านการประมวลผลภาษาธรรมชาติสำหรับภาษาไทยนั้น การตัดคำจะเป็นงานขั้นตอนแรกที่จะต้องมีการทำก่อนที่จะนำไปประมวลผลอื่นๆ ต่อไป แต่ในบางงานนอกจากจะต้องตัดคำแล้ว ยังต้องการข้อมูลเพิ่มเติมของคำนั้นเช่น หน้าที่คำ หรือ ความหมาย เป็นต้น และสำหรับขั้นตอนวิธีของการตัดคำที่ใช้ในวิทยานิพนธ์นี้จะต้องมีการนำหน้าที่คำเข้ามาช่วยในการประมวลผลด้วย ดังนั้นในบทนี้จะอธิบายถึงวิธีการกำกับหน้าที่ของคำ (Part-of-Speech Tagging)

การกำกับหน้าที่ของคำนั้นมีอยู่หลายแนวคิด ได้แก่ แนวคิดการใช้กฎ (Rule-based Approaches) แนวคิดการใช้สถิติ (Statistic-based Approaches) ซึ่งสำหรับงานด้านการประมวลผลภาษาธรรมชาตินั้น นิยมแนวคิดทางด้านสถิติมากกว่า เนื่องจากสามารถรองรับข้อมูลในหลายๆ รูปแบบได้โดยไม่ทำให้เกิดข้อผิดพลาดขึ้น และยังสามารถที่จะคำนวณค่าสถิติต่างๆ ที่นำมาใช้ได้โดยอัตโนมัติ ส่วนแนวคิดการใช้กฎจะต้องมีการใช้มนุษย์วิเคราะห์เพื่อสร้างกฎให้ครอบคลุมภาษาที่ใช้ทั้งหมด ซึ่งจะเป็นเรื่องที่ยุ่งยากมาก และยิ่งมีจำนวนกฎเกณฑ์มากขึ้นก็จะยิ่งทำให้เกิดความกำกวมมากขึ้นตามไปด้วย แต่ในปัจจุบันได้มีการพัฒนาแนวคิดการใช้กฎให้สามารถมีการสร้างกฎขึ้นมาได้เอง โดยสามารถจะสรุปกฎจากคลังข้อความที่มีอยู่ได้ แต่อย่างไรก็ตามวิธีการที่พัฒนาขึ้นมานี้ยังทำงานได้ช้า ทำให้งานต่อมาได้มีการปรับปรุงเพื่อที่จะเพิ่มความเร็วในการทำงาน

เพื่อที่จะหาวิธีการที่จะนำมาใช้ในการแก้ไขปัญหานั้น ในบทนี้จะอธิบายถึงลักษณะปัญหาการกำกับหน้าที่คำซึ่งจะได้กล่าวในส่วนถัดไป

3.1 ลักษณะปัญหาของการกำกับหน้าที่คำ

จากนิยามของการกำหนดหน้าที่คำที่ได้กล่าวไปแล้วในตอนต้น สามารถกำหนดให้เป็นสมการได้ดังสมการ 3-1

$$\tau = \max_{C_1,...,C_t} \arg PROB(C_1,...,C_t \mid w_1,...,w_t) \qquad (3\text{-}1)$$

โดยที่ τ คือ $C_1,...C_t$ ที่ทำให้ค่าความน่าจะเป็นตามสมการที่ 3-1 มีค่ามากที่สุด C_t คือหน้าที่คำของคำ w_t ส่วน $w_1, w_2, ..., w_t$ คือลำดับของคำในประโยคหนึ่งๆ และ $C_1, C_2, ..., C_t$ คือลำดับของหน้าที่คำในประโยคนั้น ส่วนความหมายจากสมการที่ 3-1 จะหมายถึงว่าภายในประโยคหนึ่งๆ ประกอบไปด้วยลำดับของคำ $w_1, w_2, .., w_t$ และให้เลือกลำดับของหน้าที่คำ $C_1, C_2, ...C_t$ ที่ทำให้ค่าความน่าจะเป็นตามสมการที่ 3-1 มีค่ามากที่สุด

3.2 วิธีการแก้ปัญหา

สำหรับการกำหนดหน้าที่ของคำที่จะนำมาใช้ในวิธีการนี้ คือนำแนวคิดการใช้สถิติเข้ามาช่วย โดยนำการกำหนดหน้าที่คำแบบไตรแกรมเข้ามาใช้

เมื่อพิจารณาจากสมการที่ 3-1 จะเห็นว่าการที่จะหาค่าความน่าจะเป็นของลำดับหน้าที่คำในสมการนี้ จำเป็นจะต้องมีคลังข้อความขนาดใหญ่มาก ซึ่งในความเป็นจริงการหาคลังข้อความขนาดดัง

กล่าวจะไม่สามารถทำได้อย่างแน่นอน ดังนั้นจึงมีการปรับปรุงสมการที่ 3-1 โดยมีการนำกฎของเบย์ (Bayes' rule) เข้ามาใช้ ซึ่งแสดงในสมการที่ 3-2

$$PROB(A \mid B) = \frac{PROB(B \mid A) \times PROB(A)}{PROB(B)} \tag{3-2}$$

ดังนั้นเมื่อนำกฎของเบย์เข้ามาปรับปรุงสมการที่ 3-1 จะได้สมการใหม่ แสดงตามสมการที่ 3-3

$$\tau = \max_{c_1,...,c_t} \arg \frac{PROB(C_1,...,C_t) \times PROB(w_1,....,w_t \mid C_1,...,C_t)}{PROB(w_1,....,w_t)} \tag{3-3}$$

จากสมการที่ 3-3 จะเห็นว่าต้องมีการคำนวณ $PROB\ (w_1,...,w_t)$ ซึ่งเป็นค่าคงที่ ดังนั้นเราจึงสามารถละค่านี้ได้ โดยไม่กระทบกับผลลัพธ์ ทำให้สมการที่ 3-3 สามารถลดรูปได้ ซึ่งแสดงในสมการที่ 3-4

$$\tau = \max_{c_1,...,c_t} \arg PROB(C_1,...,C_t) \times PROB(w_1,....,w_t \mid C_1,...,C_t) \tag{3-4}$$

เมื่อทำการลดรูปจากสมการที่ 3-1 มาเป็นสมการที่ 3-4 แล้ว ยังจำเป็นต้องการคลังข้อความจำนวนมากเช่นกัน แต่อย่างไรก็ตามสมการนี้สามารถที่จะทำการคำนวณโดยประมาณได้ ซึ่งจะทำให้การคำนวณสามารถทำให้ง่ายขึ้น และจำนวนคลังข้อความที่จะนำมาใช้นั้นมีขนาดลดลงอย่างมาก โดยสร้างสมมุติฐานว่าหน้าที่ของคำหนึ่งๆ จะขึ้นอยู่กับหน้าที่คำของคำก่อนหน้า 1 คำ หรือ 2 คำ ซึ่งสามารถเรียกได้ว่าเป็นแบบไบแกรม (Bigram) หรือ ไตรแกรม (Trigram) ตามลำดับ

สำหรับการกำกับหน้าที่คำที่จะนำมาใช้ในวิทยานิพนธ์นี้ จะนำโมเดลไตรแกรมเข้ามาใช้ ดังนั้นการคำนวณค่า $PROB(C_1,..,C_t)$ จะสามารถคำนวณได้ดังสมการที่ 3-5

$$PROB(C_1,...,C_t) \cong \prod_{i=1}^{t} PROB(C_i \mid C_{i-1}, C_{i-2}) \tag{3-5}$$

ส่วนการคำนวณค่า $PROB(w_1,...,w_t \mid C_1,...,C_t)$ ในสมการที่ 3-4 สามารถจะประมาณ โดยสมมุติว่าหน้าที่ของคำหนึ่งคำจะไม่ขึ้นอยู่กับคำก่อนหน้า หรือคำที่ตามหลัง ดังนั้นการคำนวณค่า $PROB(w_1,...,w_t \mid C_1,...,C_t)$ สามารถจะประมาณได้ดังสมการ 3-6

$$PROB(w_1,....,w_t \mid C_1,...,C_t) \cong \prod_{i=1}^{t} PROB(w_i \mid C_i) \tag{3-6}$$

ดังนั้นจากสมการที่ 3-4 สามารถจะประมาณตามสมการที่ 3-5 และ 3-6 ได้ดังสมการที่ 3-7

$$\tau = \max_{C_1,\ldots,C_t} \arg \prod_{i=1}^{t} PROB(C_i \mid C_{i-1}, C_{i-2}) \times PROB(w_i \mid C_i) \qquad (3\text{-}7)$$

เมื่อได้สมการในการหาลำดับของหน้าที่คำ ดังสมการ 3-7 แล้ว จะเห็นว่าคลังข้อความที่จะนำมาใช้ในการเก็บค่าสถิตินั้นจะมีขนาดน้อยลง ทำให้ในความเป็นจริงการหาคลังข้อความที่มีขนาดเพียงพอที่จะสามารถนำมาใช้ตามสมการ 3-7 นั้นเป็นไปได้จริง ดังนั้นในวิทยานิพนธ์นี้ก็จะนำสมการนี้มาใช้ในการกำกับหน้าที่คำ แต่ในความเป็นจริงถ้าเขียนโปรแกรมจากสมการนี้ตามตรงจะมีการคำนวณจำนวนมาก ทำให้โปรแกรมทำงานได้ช้า ดังนั้นจึงต้องมีการปรับปรุงโดยนำเทคนิคเรื่องไดนามิคโปรแกรมมิ่ง (Dynamic Programming) เข้ามาช่วย ส่วนในรายละเอียดนั้น จะทำการอธิบายในส่วนถัดไป

3.3 การเพิ่มประสิทธิภาพ

เนื่องจากเมื่อมีการเขียนโปรแกรมกำกับหน้าที่คำ โดยใช้สมการที่ 3-7 ขึ้นโดยตรงนั้นจะทำให้โปรแกรมได้ช้ามาก เนื่องจากถ้านำประโยคที่ตัดคำแล้วมากำกับหน้าที่คำ ซึ่งประกอบไปด้วยจำนวนคำ T คำ และจำนวนหน้าที่คำสามารถแปลงออกได้เป็น N หมวดหมู่ ในกรณีที่แย่ที่สุดคือคำหนึ่งคำสามารถมีหน้าที่คำได้ทั้งหมด N หมวดหมู่ ดังนั้นในการคำนวณตามสมการที่ 3-7 จะต้องใช้การคำนวณประมาณ kxN^T โดยค่า k คือค่าคงที่ ซึ่งจะเห็นว่าวิธีการนี้จะใช้เวลาค่อนข้างมากโดยจะขึ้นอยู่กับจำนวนคำในประโยคที่จะนำมากำกับหน้าที่คำ โดยเวลาที่ใช้นั้นจะเป็นสัดส่วนแบบเอกซ์โปเนนเชียล (Exponential) ซึ่งจะยิ่งช้ามากถ้าจำนวนคำในประโยคมาก ดังนั้นจึงมีการพัฒนาโดยนำเทคนิคเรื่องไดนามิคโปรแกรมมิ่งเข้ามาช่วย และขั้นตอนวิธีที่นำมาใช้ในการปรับปรุงความเร็วนี้มีชื่อเรียกว่า ขั้นตอนวิธีวิเทอร์บิ (Viterbi Algorithm) เมื่อนำขั้นตอนวิเทอร์บิเข้ามาประยุกต์ใช้กับการกำกับหน้าที่คำแบบไตรแกรม ซึ่งแสดงได้ตามรูปที่ 3-1

ขั้นตอนวิธีวิเทอร์บิที่แสดงในรูปที่ 3-1 นั้น จะมีการสร้างแถวลำดับ (Array) ขนาด $NxNxT$ จำนวน 2 ชุดโดย N คือจำนวนหน้าที่คำที่เป็นไปได้ทั้งหมด และ T คือจำนวนคำในประโยคที่จะนำมากำกับหน้าที่คำ โดยที่แถวลำดับชุดแรกคือแถวลำดับ *seqscore*[i][j][t] จะทำการเก็บค่าความน่าจะเป็นที่ดีที่สุดของการกำกับหน้าที่คำของ $w_1, \ldots, w_t$ ซึ่งคำที่หน้าที่ของคำ w_t กับ w_{t-1} จะมีหน้าที่คำเป็น L_i และ L_j ตามลำดับ ส่วนแถวลำดับชุดที่สองคือ *backptr*[i][j][t] จะเก็บหน้าที่คำของคำ $t-2$ เมื่อคำที่ t และ t-1 มีหน้าที่คำเป็น L_j และ L_i ตามลำดับ

กำหนดให้ $w_1,..,w_t$ เป็นลำดับคำในประโยค $L_1,\ldots, L_n$ เป็นหน้าที่คำที่เป็นไปได้ $Prob(w_t \mid L_i)$คือค่าความน่าจะเป็นของคำศัพท์ w_t เมื่อกำหนดให้มีหน้าที่คำเป็น L_i และค่าความน่าจะเป็นของไตรแกรมคือ $Prob\ (\ L_k \mid L_i,L_j\ )$ ดังนั้นให้หาลำดับของหน้าที่คำ $C_1,..,C_T$ ที่เป็นของลำดับคำในประโยคที่มีความน่าจะเป็นมากที่สุด

Initialization Step

for i =1 to N do

for j=1 to N do

seqscore[i][j][1] = Prob($W_1|L_i$) × Prob($L_i \mid \phi$) × Prob($W_2 \mid L_j$)
× Prob($Lj \mid Li, \phi$)

backptr[i] [j] [2] = 0

Iteration Step

for t=3 to T do

for j=1 to N do

for k=1 to N do

seqscore[j][k][t] = max $_{i=1,N}$ (seqscore[i][j][t-1]
× Prob($L_k|L_j,L_i$)) × Prob($W_t|L_k$)

backptr[j][k][t] = ค่า i ที่ทำให้ค่าสมการที่ผ่านมาเป็นค่าที่มากที่สุด

Sequence Identification Step

C[T] = k and C[T-1] = j โดยที่ j และ k นั้นทำให้ seqscore[j][k][T] มีค่ามาที่สุด

for i=T-2 to 1 do

C[i] = backptr [C[i+1]] [C[i+1]] [i+1]

รูปที่ 3-1 ขั้นตอนวิธีวิเทอร์บิ (Viterbi Algorithm)

จากการคำนวณหาหน้าที่คำโดยใช้ขั้นตอนวิธีวิเทอรบินั้นจะสามารถลดเวลาการคำนวนได้ โดยจากของเดิมที่ต้องใช้เวลาเป็นลัดส่วนกับ kN^T ส่วนวิธีการนี้ก็จะใช้เวลาเป็นลัดส่วน N^3T ดังนั้นจะเห็นว่าเมื่อนำขั้นตอนวิธีวิเทอร์บิเข้ามาใช้นั้นในการกำกับหน้าที่คำจะสามารถลดเวลาได้เป็นจำนวนมาก