

บทที่ 6

การสังเคราะห์คำและประโยคของภาษาไทย

6.1 การสังเคราะห์โครงสร้างประโยคภาษาไทย

ทุกประโยคที่มีคำกริยา ไม่ว่าจะ เป็นคำกริยาในประโยคหลัก หรือประโยคย่อย จะต้องผ่านกฎไวยากรณ์ข้างล่าง เพื่อย้ายคำกริยาไปไว้ด้านขวาของคำทุกคำในประโยค เพื่อเป็นประโยชน์ในการสร้างประโยคภาษาไทย

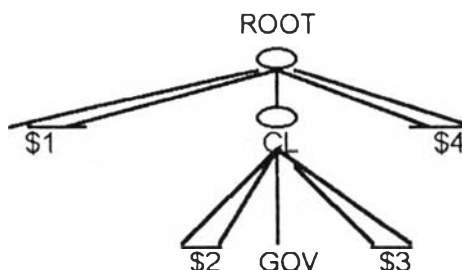
```
(GS-RULE* CL-GOV->RIGHT.  
($1 (@CL $2 GOV $3) $4)  
( (CL (& KLOCK (MEM K '(VCL SCL PARTCL RELCL))))  
(GOV (& GOV ^GOVLOCK CAT)) )  
($1 (CL $2 $3 GOV) $4)  
( (GOV (:= GOV GOVLOCK))) )
```

ในกฎไวยากรณ์นี้

ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

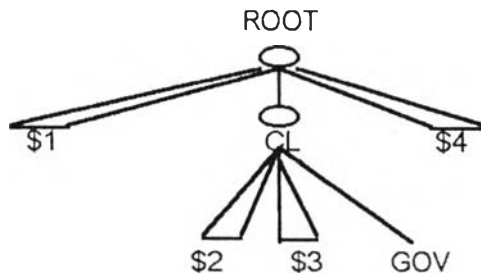
ชื่อกฎไวยากรณ์ = CL-GOV->RIGHT

รูปแบบต้นไม้เข้า = (\$1 (@CL \$2 GOV \$3) \$4) มีรูปแบบเป็น



เงื่อนโซ่ = ถ้าประโยคมีคำกริยา

รูปแบบต้นไม้เอก = (\$1 (CL \$2 \$3 GOV) \$4) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

GOV = Lock ประโยคเพื่อไม่ให้ทำซ้ำในประโยคนี้อีก

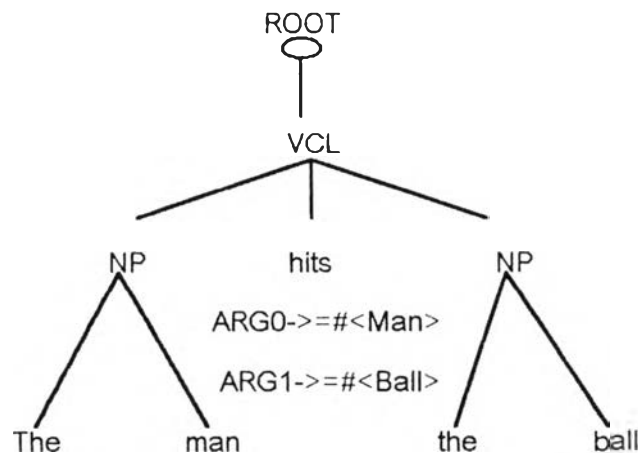
ในบทที่ 3 ได้แสดงโครงสร้างประโยคพื้นฐานและแสดงความแตกต่างของโครงสร้างประโยคภาษาอังกฤษและภาษาไทย ซึ่งสามารถถ่ายทอดโครงสร้างประโยคภาษาอังกฤษเป็นโครงสร้างประโยคภาษาไทย ได้เป็นข้อ ๆ ดังต่อไปนี้

5.2.1 ประโยคพื้นฐานแบบที่ 1 ตัวอย่างประโยคเช่น

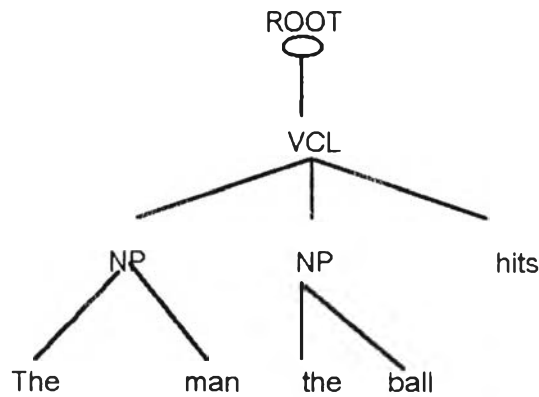
The man hits the ball.

ผู้ชาย ตี ลูกบอล (โครงสร้างภาษาไทย)

เมื่อผ่านขั้นตอนการวิเคราะห์โครงสร้างภาษาอังกฤษแล้ว จะได้โครงสร้างต้นไม้และลักษณะทางภาษาดังนี้



จะเห็นได้ว่าระบบได้วิเคราะห์ไว้ว่า คำว่า Man เป็น Argument 0 และ Ball เป็น Argument 1 และเมื่อผ่านกฎไวยากรณ์ CL-GOV->RIGHT จะได้โครงสร้างต้นไม้ดังนี้



เราสามารถเขียนกฎไวยากรณ์ให้เป็นประโยคโครงสร้างภาษาไทยได้ดังนี้

```

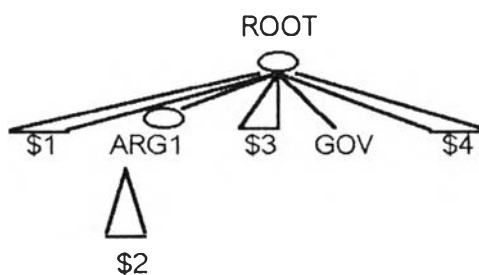
(GS-RULE* CL-ARG1=ACT
($1 (@ARG1 $2) $3 GOV $4)
( (ROOT (& KLOCK ^PASSIVE (MEM K '(VCL SCL PARTCL RELCL))))
  (ARG1 (& ARG1 ^SFLOCK (N== R 'RREL)))
  (GOV (& GOV CAT)) )
($1 $3 GOV $4 (ARG1 $2))
( (ARG1 (:= OBJ1 SFLOCK))))
  
```

ในกฎไวยากรณ์นี้

ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

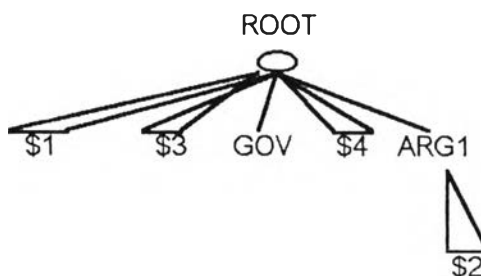
ชื่อกฎไวยากรณ์ = CL-ARG1=ACT

รูปแบบต้นไม้เข้า = (\$1 (@ARG1 \$2) \$3 GOV \$4) มีรูปแบบเป็น



เงื่อนไข = ถ้าประโยคไม่เป็นประโยค Passive Voice และมีนามวลีที่มีความสัมพันธ์ในประโยคเป็น ARG1 ตามด้วยคำกริยา

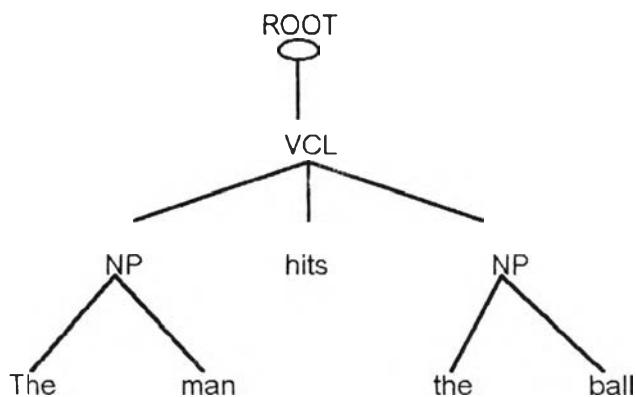
รูปแบบต้นไม้เอก = (\$1 \$3 GOV \$4 (ARG1 \$2)) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

ARG1 = มีหน้าที่ในประโยคคือ เป็น กรรมตรง (Object1)

เมื่อผ่านกฎไวยากรณ์นี้จะได้โครงรูปต้นไม้ของประโยค The man hits the ball. เป็น

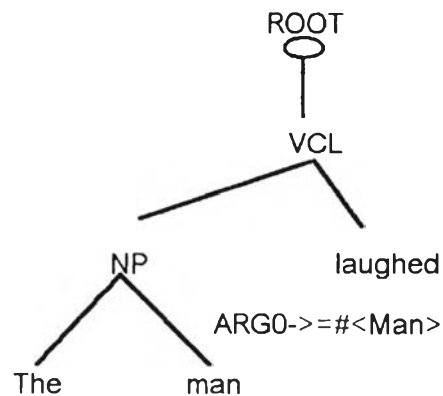


5.2.2 ประโยคพื้นฐานแบบที่ 2 ตัวอย่างประโยคเช่น

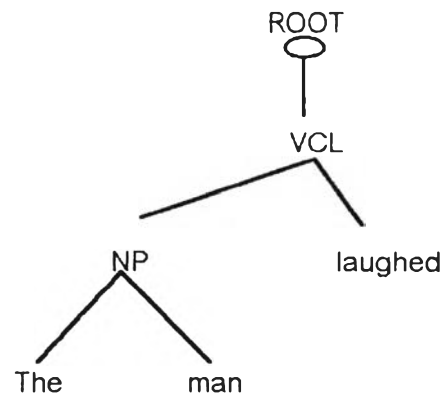
The man laughed

ผู้ชาย หัวเราะ (โครงสร้างภาษาไทย)

เมื่อผ่านขั้นตอนการวิเคราะห์โครงสร้างภาษาอังกฤษแล้ว จะได้โครงสร้างต้นไม้และลักษณะทางภาษาดังนี้



จะเห็นได้ว่าระบบได้วิเคราะห์ไว้ว่า คำว่า Man เป็น Argument 0 และเมื่อผ่านกฎไวยากรณ์ CL-GOV->RIGHT จะได้โครงสร้างต้นไม้ดังนี้



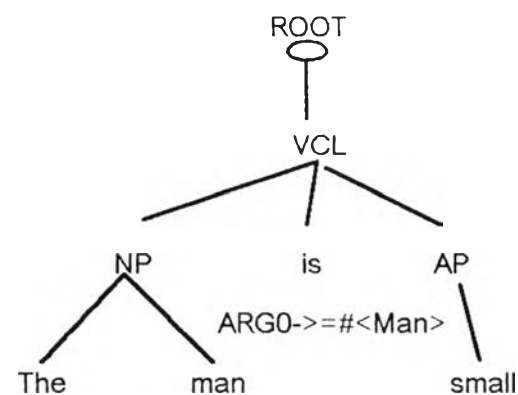
ไม่จำเป็นต้องมีกฎไวยากรณ์อีก เพราะเป็นไปตามโครงสร้างภาษาไทยแล้ว

5.2.3 ประโยคพื้นฐานแบบที่ 3 ตัวอย่างประโยคเช่น

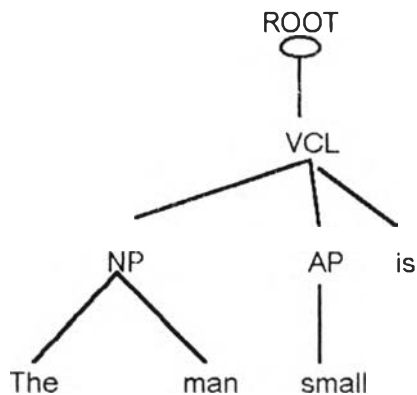
The man is small

ผู้ชาย ตัวเล็ก (โครงสร้างภาษาไทย)

เมื่อผ่านขั้นตอนการวิเคราะห์โครงสร้างภาษาอังกฤษแล้ว จะได้โครงสร้างต้นไม้และลักษณะทางภาษาดังนี้



จะเห็นได้ว่าระบบได้วิเคราะห์ไว้ว่า คำว่า Man เป็น Argument 0 และ small เป็น Transfer to ARG0 และเมื่อผ่านกฎไวยากรณ์ CL-GOV->RIGHT จะได้โครงสร้างต้นไม้ดังนี้



เราสามารถเขียนกฎไวยากรณ์ให้เป็นประโยคโครงสร้างภาษาไทยได้ดังนี้

```

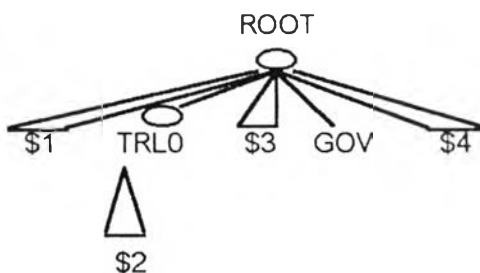
(GS-RULE* CL-TRL0=COP
($1 (@TRL0 $2) $3 GOV $4)
((ROOT (& COP KLOCK (MEM K '(VCL SCL PARTCL RELCL))))
(TRL0 (& TRL0 ^SFLOCK))
(GOV (& COP GOV CAT))
($1 $3 GOV $4 (TRL0 $2))
((TRL0 (:= ATSUBJ SFLOCK) )))
  
```

ในกฎไวยากรณ์นี้

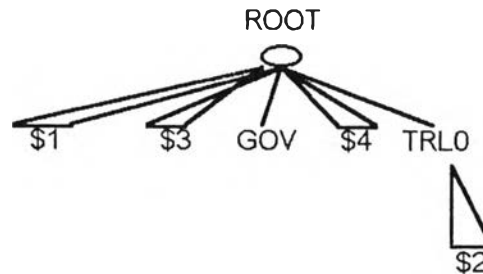
ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

ชื่อกฎไวยากรณ์ = CL-TRL0=COP

รูปแบบต้นไม้เข้า = (\$1 (@TRL0 \$2) \$3 GOV \$4) มีรูปแบบเป็น



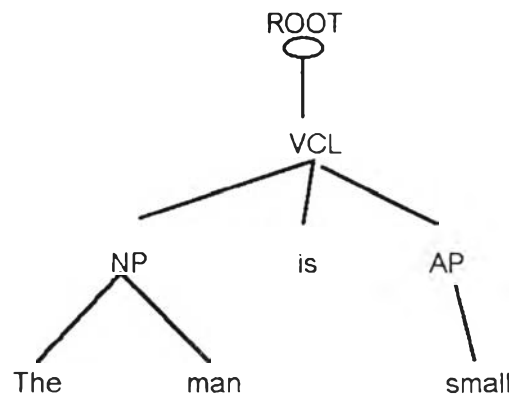
เงื่อนไข = ถ้าประโยคมีนามวลที่มีความสัมพันธ์ในประโยคเป็น TRLO ตามด้วยคำกริยา
 รูปแบบต้นไม้ = (\$1 \$3 GOV \$4 (TRLO \$2)) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

TRLO = มีหน้าที่ในประโยคคือเป็น คำขยายของประธานของประโยค (attribute of subject)

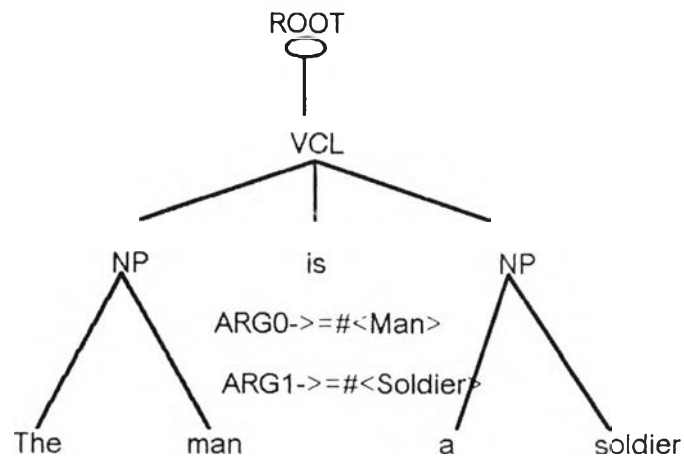
เมื่อผ่านกฎไวยากรณ์นี้จะได้โครงรูปต้นไม้ของประโยค The man is small. เป็น



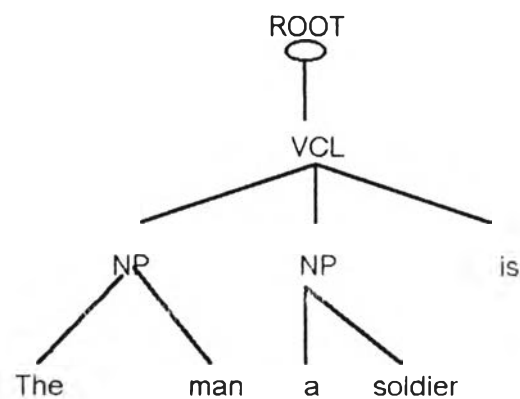
5.2.4 ประโยคพื้นฐานแบบที่ 4 ตัวอย่างประโยคเช่น

The man is a soldier.
 ผู้ชาย เป็น ทหาร (โครงสร้างภาษาไทย)

เมื่อผ่านขั้นตอนการวิเคราะห์โครงสร้างภาษาอังกฤษแล้ว จะได้โครงสร้างต้นไม้และ
 ลักษณะทางภาษาดังนี้



จะเห็นได้ว่าระบบได้วิเคราะห์ไว้ว่า คำว่า Man เป็น Argument 0 และ soldier เป็น Argument 1 และเมื่อผ่านกฎไวยากรณ์ CL-GOV->RIGHT จะได้โครงสร้างต้นไม้ดังนี้



เราสามารถเขียนกฎไวยากรณ์ให้เป็นประโยคโครงสร้างภาษาไทยได้ดังนี้

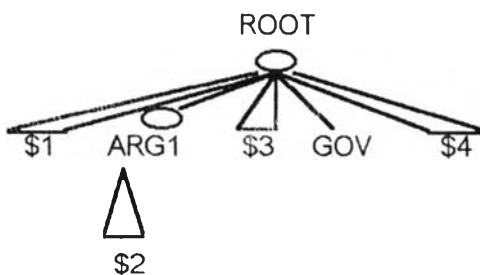
```
(GS-RULE* CL-ARG1=ACT
($1 (@ARG1 $2) $3 GOV $4)
( (ROOT (& KLOCK ^PASSIVE (MEM K '(VCL SCL PARTCL RELCL))))
(ARG1 (& ARG1 ^SFLOCK (N== R 'RREL)))
(GOV (& GOV CAT)) )
($1 $3 GOV $4 (ARG1 $2))
( (ARG1 (:= OBJ1 SFLOCK))))
```


ในกฎไวยากรณ์นี้

ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

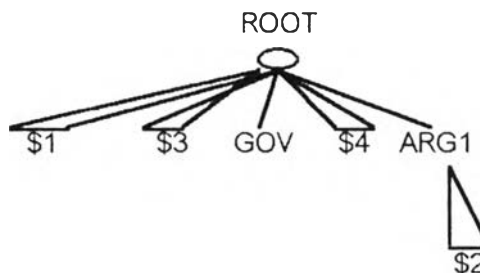
ชื่อกฎไวยากรณ์ = CL-ARG1=ACT

รูปแบบต้นไม้เข้า = (\$1 (@ARG1 \$2) \$3 GOV \$4) มีรูปแบบเป็น



เงื่อนไข = ถ้าประโยคไม่เป็นประโยค Passive Voice และมีนามวลีที่มีความสัมพันธ์ในประโยคเป็น ARG1 ตามด้วยคำกริยา

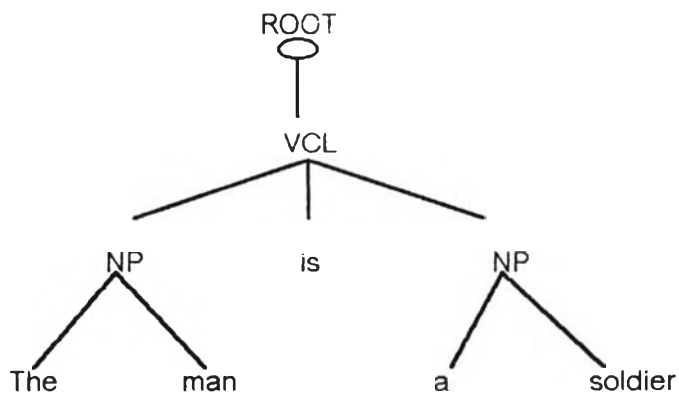
รูปแบบต้นไม้ออก = (\$1 \$3 GOV \$4 (ARG1 \$2)) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

ARG1 = มีหน้าที่ในประโยคคือ เป็น กรรมตรง (Object1)

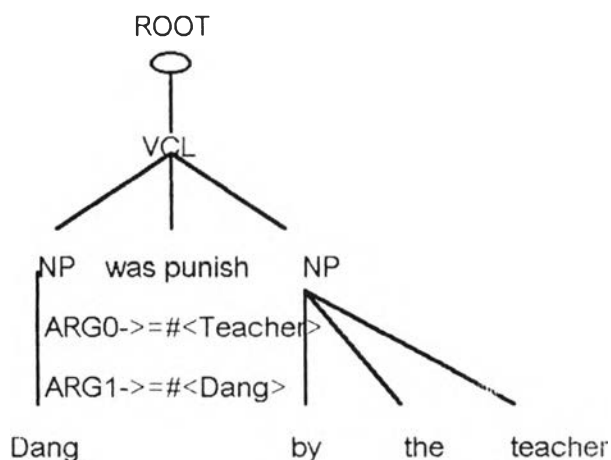
เมื่อผ่านกฎไวยากรณ์นี้จะได้โครงรูปต้นไม้ของประโยค The man is a soldier. เป็น



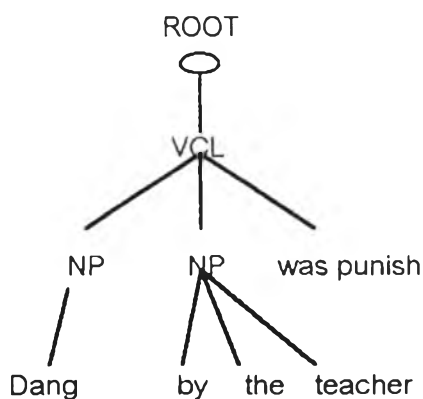
5.2.5 ทำประโยค Passive Voice เป็นโครงสร้างของภาษาไทย ตัวอย่างประโยคเช่น

Dang was punished by the teacher
 แดง ถูก ลงโทษ โดย ครู
 แดง ถูก ครู ลงโทษ(โครงสร้างภาษาไทย)

เมื่อผ่านขั้นตอนการวิเคราะห์โครงสร้างภาษาอังกฤษแล้ว จะได้โครงสร้างต้นไม้และลักษณะทางภาษาดังนี้



จะเห็นได้ว่าระบบได้วิเคราะห์ไว้ว่า คำว่า Teacher เป็น Argument 0 และ Dang เป็น Argument 1 และมีรูปประโยคเป็นแบบ Passive Voice และเมื่อผ่านกฎไวยากรณ์ CL-GOV->RIGHT จะได้โครงสร้างต้นไม้ดังนี้



เราสามารถเขียนกฎไวยากรณ์ให้เป็นประโยคโครงสร้างภาษาไทยได้ดังนี้

```

(GS-RULE* CL-ARG0=PAS
($1 (@ARG0 BY $2) $3 GOV $4)
( (ROOT (& PASSIVE KLOCK (MEM K '(VCL SCL PARTCL RELCL))))
  (ARG0 (& ARG0 ^SFLOCK ^RREL))
  (GOV (& GOV CAT)) )
($1 Toog (ARG0 BY $2) $3 GOV $4)
( (ARG0 (:= CIRC SFLOCK) )
  (Toog (:= (BUL "ถูก")))
  (BY (:= (BUL "")))
  (GOV (:= (VAFFIX ""))) ) )

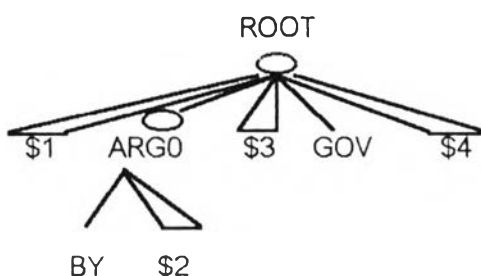
```

ในกฎไวยากรณ์นี้

ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

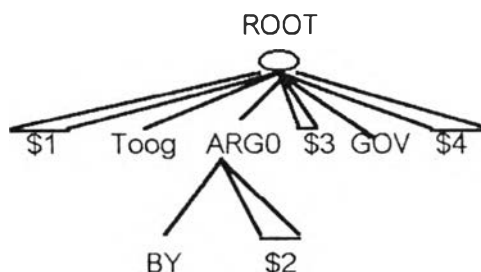
ชื่อกฎไวยากรณ์ = CL-ARG0=PAS

รูปแบบต้นไม้ออก = (\$1 (@ARG0 BY \$2) \$3 GOV \$4) มีรูปแบบเป็น



เงื่อนไข = ถ้าประโยคมีรูป Passive Voice และมีนามวลีที่มีความสัมพันธ์ในประโยคเป็น ARG0 ตามด้วยคำกริยา

รูปแบบต้นไม้ออก = (\$1 Toog (ARG0 BY \$2) \$3 GOV \$4) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

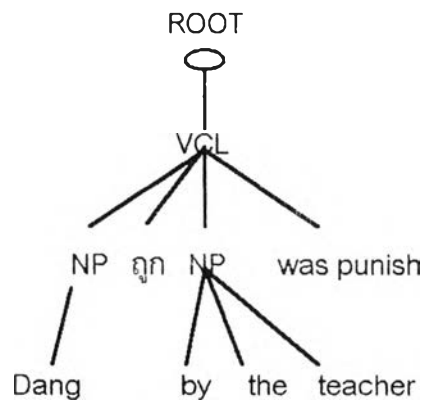
ARG0 = มีหน้าที่ในประโยคคือ เป็นส่วนประกอบแวดล้อมของประโยค (Circumstantial)

TOOG = เพิ่มคำว่า ถูก ในประโยค

BY = ไม่มีคำแปลภาษาไทย

GOV = ไม่มี AFFIX

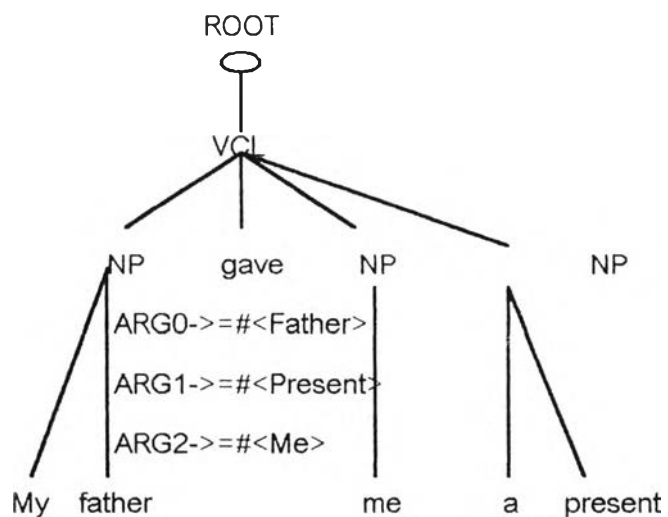
เมื่อผ่านกฎไวยากรณ์นี้จะได้โครงรูปต้นไม้ของประโยค Dang was punished by the teacher.
เป็น



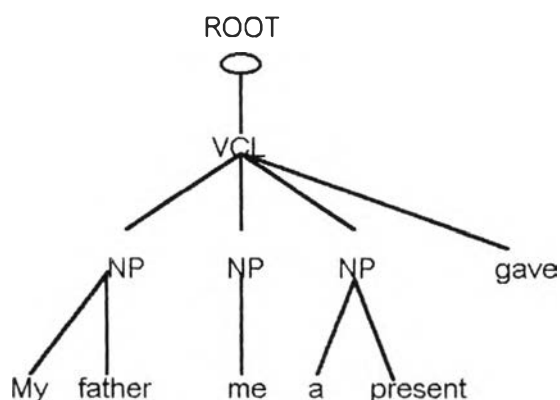
5.2.6 ทำประโยคภาษาอังกฤษโครงรูปแบบที่ 1 ที่มีกรรมตรง และกรรมรอง ให้เป็นโครงสร้างประโยคภาษาไทย ตัวอย่างประโยคเช่น

My	father	gave	me	a	present
พ่อ	ของฉัน	ให้	ฉัน	ของขวัญ	
พ่อ	ของฉัน	ให้	ของขวัญ	กับ	ฉัน (โครงสร้างภาษาไทย)

เมื่อผ่านขั้นตอนการวิเคราะห์โครงสร้างภาษาอังกฤษแล้ว จะได้โครงสร้างต้นไม้และลักษณะทางภาษาดังนี้



จะเห็นได้ว่าระบบได้วิเคราะห์ไว้ว่า คำว่า Father เป็น Argument 0 present เป็น Argument 1 และ me เป็น Argument 2 และเมื่อผ่านกฎไวยากรณ์ CL-GOV->RIGHT จะได้โครงสร้างต้นไม้ดังนี้



เราสามารถเขียนกฎไวยากรณ์ 2 กฎไวยากรณ์ให้เป็นประโยคโครงสร้างภาษาไทยได้ดังนี้
กฎไวยากรณ์แรก

```

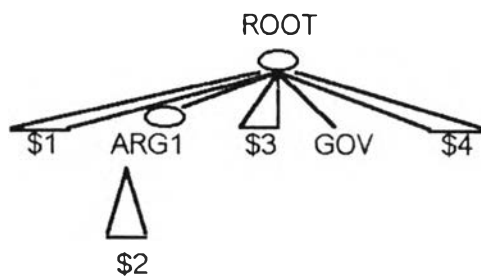
(GS-RULE* CL-ARG1=ACT
($1 (@ARG1 $2) $3 GOV $4)
((ROOT (& KLOCK ^PASSIVE (MEM K '(VCL SCL PARTCL RELCL))))
(ARG1 (& ARG1 ^SFLOCK (N== R 'RREL)))
(GOV (& GOV CAT)))
($1 $3 GOV $4 (ARG1 $2))
((ARG1 (:= OBJ1 SFLOCK))))
  
```

ในกฎไวยากรณ์นี้

ชื่อชั้นตอน = GS-RULE หมายถึง ชั้นสร้างโครงสร้างประโยค (structural generation)

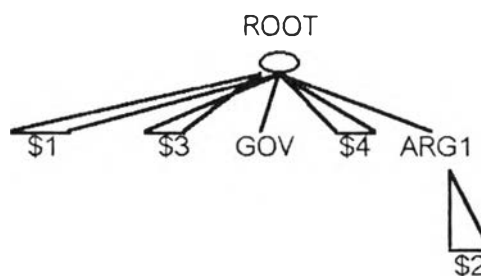
ชื่อกฎไวยากรณ์ = CL-ARG1=ACT

รูปแบบต้นไม้เข้า = (\$1 (@ARG1 \$2) \$3 GOV \$4) มีรูปแบบเป็น



เงื่อนไข = ถ้าประโยคไม่เป็นประโยค Passive Voice และมีนามวลที่มีความสัมพันธ์ในประโยคเป็น ARG1 ตามด้วยคำกริยา

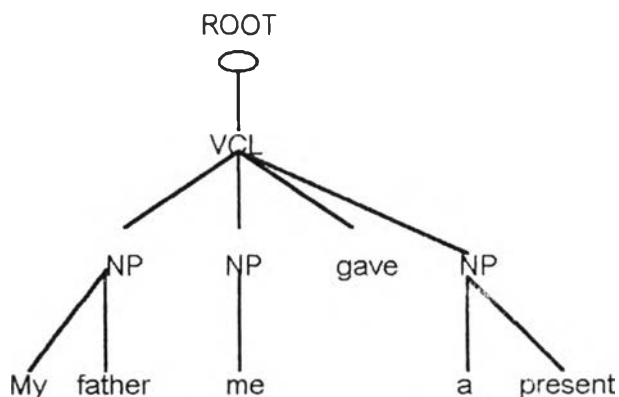
รูปแบบต้นไม้ ออก = (\$1 \$3 GOV \$4 (ARG1 \$2)) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

ARG1 = มีหน้าที่ในประโยคคือ เป็น กรรมตรง (Object1)

เมื่อผ่านกฎไวยากรณ์นี้จะได้โครงรูปต้นไม้ของประโยค My father gave me a present. เป็น



กฎไวยากรณ์ที่สอง

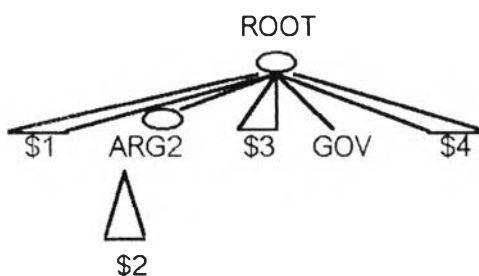
```
(GS-RULE* CL-ARG2=ACT
($1 (@ARG2 $2) $3 GOV $4)
( (ROOT (& KLOCK ^PASSIVE (MEM K '(VCL SCL PARTCL RELCL))))
  (ARG2 (& ARG2 ^SFLOCK ^RREL))
  (GOV (& GOV CAT)) )
($1 $3 GOV $4 (ARG2 $2))
( (ARG2 (:= OBJ2 SFLOCK) )))
```

ในกฎไวยากรณ์นี้

ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

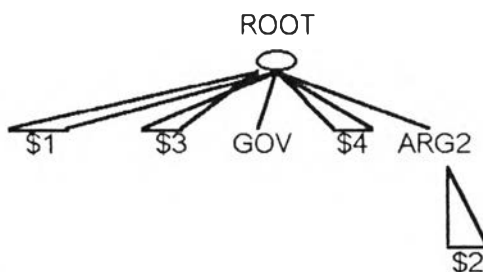
ชื่อกฎไวยากรณ์ = CL-ARG2=ACT

รูปแบบต้นไม้เข้า = (\$1 (@ARG2 \$2) \$3 GOV \$4) มีรูปแบบเป็น



เงื่อนไข = ถ้าประโยคไม่เป็นประโยค Passive Voice และมีนามวลีที่มีความสัมพันธ์ในประโยคเป็น ARG2 ตามด้วยคำกริยา

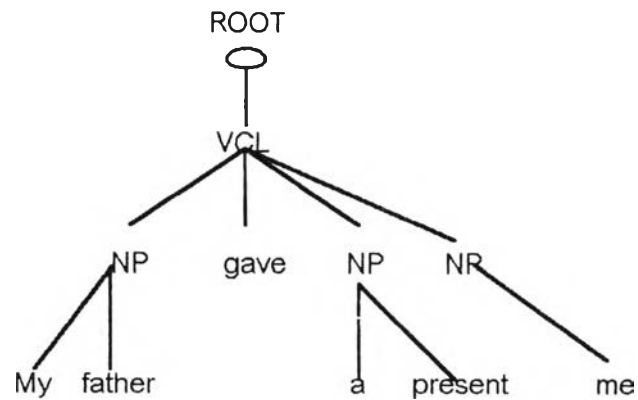
รูปแบบต้นไม้ออก = (\$1 \$3 GOV \$4 (ARG2 \$2)) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

ARG2 = มีหน้าที่ในประโยคคือ เป็น กรรมรอง (Object2)

เมื่อผ่านกฎไวยากรณ์นี้จะได้โครงรูปต้นไม้ของประโยค My father gave me a present. เป็น



5.2.7 ใส่วลีลักษณะนามของภาษาไทย ตัวอย่างประโยคเช่น

I	want	two	oranges	
ฉัน	ต้องการ	สอง	ส้ม	
ฉัน	ต้องการ	ส้ม	สอง	ผล (โครงสร้างภาษาไทย)

ในกรณีนี้เป็นการจัดการกับนามวลี ซึ่งในตัวอย่างคือ Two oranges

เราสามารถเขียนกฎไวยากรณ์ 2 กฎไวยากรณ์ให้เป็นประโยคโครงสร้างภาษาไทยได้ดัง

นี้

กฎไวยากรณ์แรก

```

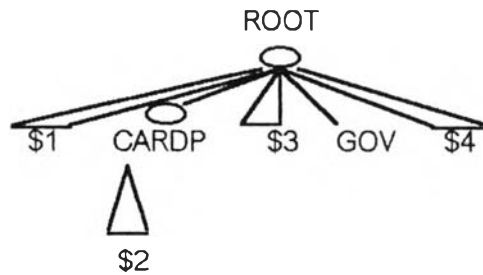
(GS-RULE*NP-CARDP=AFTER
($1 (@CARDP $2) $3 GOV $4)
( (ROOT (& NP KLOCK))
(CARDP (& (OR CARDP) ^SFLOCK))
(GOV (& GOV GOVLOCK) ) )
($1 $3 GOV $4 (CARDP $2))
( (CARDP (:= ATG SFLOCK KLOCK) ) ) )
  
```

ในกฎไวยากรณ์นี้

ข้อขึ้นตอน = GS-RULE หมายถึง ชั้นสร้างโครงสร้างประโยค (structural generation)

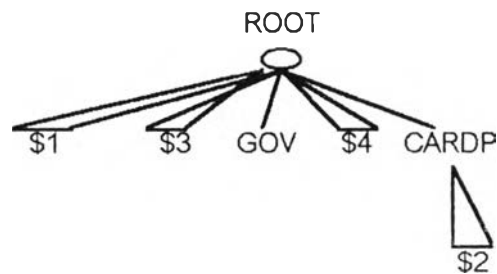
ข้อกฎไวยากรณ์ = NP-CARDP=AFTER

รูปแบบต้นไม้เข้า = (\$1 (@CARDP \$2) \$3 GOV \$4) มีรูปแบบเป็น



เงื่อนไข = ถ้านามวลีมีค่าที่มีความหมายเป็นตัวเลข ตามด้วยคำนาม

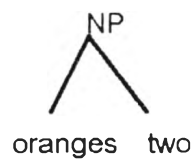
รูปแบบต้นไม้ออก = (\$1 \$3 GOV \$4 (CARDP \$2)) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

CARDP = มีหน้าที่ในประโยคคือ เป็น คำขยายของคำหลักของประโยค (attribute of governor)

เมื่อผ่านขั้นตอนนี้ จะมีต้นไม้ย่อยของนามวลี คือ



กฎไวยากรณ์ที่สอง

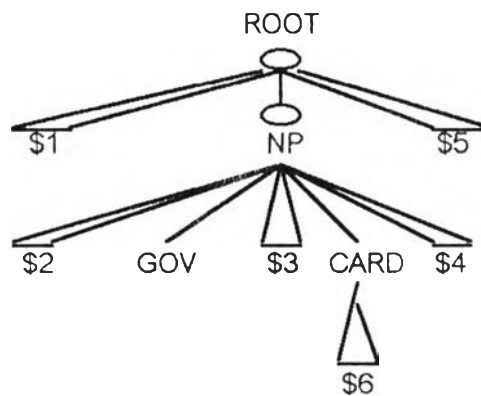
```
(GS-RULE* NP=PENBIL
($1 (@NP $2 GOV $3 (CARDP $6) $4) $5)
( (NP (& NP KLOCK))
(CARDP (OR CARDP (& DQTF (n== bul ("บางคน")) )))
(GOV (& GOV ^PENBIL CAT)) )
($1 (NP $2 GOV $3 (CARDP $6) CLF $4) $5)
( (GOV (:= PENBIL))
(CLF (:= (BUL (GOV BUL-PENBIL))))
(CLF (:> (BUL (GOV BUL)))) ) )
```

ในกฎไวยากรณ์นี้

ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

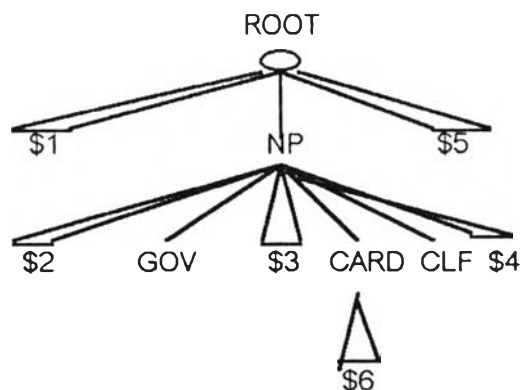
ชื่อกฎไวยากรณ์ = NP=PENBIL

รูปแบบต้นไม้เข้า = (\$1 (@NP \$2 GOV \$3 (CARDP \$6) \$4) \$5) มีรูปแบบเป็น



เงื่อนไข = ถ้านามวลีมีค่านาม ตามด้วยคำที่มีความหมายเป็นตัวเลข

รูปแบบต้นไม้ออก = (\$1 (NP \$2 GOV \$3 (CARDP \$6) CLF \$4) \$5) มีรูปแบบเป็น

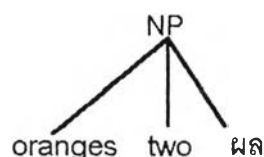


ลักษณะทางภาษาเพิ่มเติม

GOV = ให้ PENBIL = T

CLF = ลักษณะนามของคำนาม (GOV) ที่ระบุไว้ใน LL-DICT ถ้าไม่ได้ระบุไว้ใน LL-DICT ก็ให้เท่ากับคำนามนั้น

เมื่อผ่านขั้นตอนนี้ จะมีต้นไม้ย่อยของนามวลี คือ



5.2.8 เรียงลำดับส่วนขยายนาม

5.2.8.1 ส่วนขยายนามที่เป็นตัวบ่งชี้ (Determiner) ประเภทที่ใช้แทนสิ่งและผู้พูด ซึ่งเฉพาะเจาะจงถึง (DDEM) เช่น this, that, those, which, these ตัวอย่างประโยค เช่น

These books are mine.

เหล่านี้ หนังสือ เป็น ของฉัน

หนังสือ เหล่านี้ เป็น ของฉัน (โครงสร้างภาษาไทย)

ในกรณีนี้เป็นการจัดการกับนามวลี ซึ่งในตัวอย่างคือ These books

เราสามารถเขียนกฎไวยากรณ์ ให้เป็นประโยคโครงสร้างภาษาไทยได้ดังนี้

```

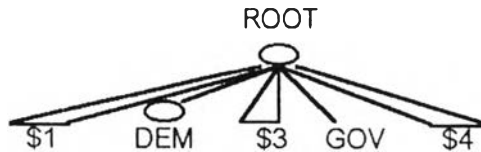
(GS-RULE NP-DEM
($1 @DEM $3 GOV $4)
( (ROOT (& NP KLOCK))
(DEM (& DDEM ^CATLOCK =D))
(GOV (& GOV GOVLOCK)) )
($1 $3 GOV $4 DEM)
( (DEM (:= DES CATLOCK)) ) )
  
```

ในกฎไวยากรณ์นี้

ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

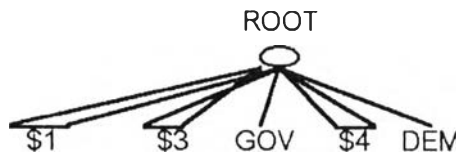
ชื่อกฎไวยากรณ์ = NP-DEM

รูปแบบต้นไม้เข้า = (\$1 @DEM \$3 GOV \$4) มีรูปแบบเป็น



เงื่อนไข = ถ้านามวลีมีคำบ่งชี้ประเภท DDEM ตามด้วยคำนาม

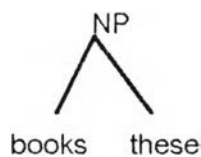
รูปแบบต้นไม้ออก = (\$1 \$3 GOV \$4 DEM) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

DEM = มีหน้าที่ในประโยคคือ เป็นคำที่ระบุแล้ว (designation)

เมื่อผ่านขั้นตอนนี้ จะมีต้นไม้ย่อยของนามวลี คือ



5.2.8.2 ส่วนขยายนามที่เป็นตัวบ่งชี้ (Determiner) ประเภทที่เป็นคำที่แทนตัวผู้พูด ผู้ที่พูดด้วย หรือผู้ที่กล่าวถึง (DPOS) เช่น our, his, her, your, my, their ตัวอย่างประโยค เช่น

This	is	my	pen.	
นี้	เป็น	ของฉัน	ปากกา	
นี้	เป็น	ปากกา	ของฉัน	(โครงสร้างภาษาไทย)

ในกรณีนี้เป็นการจัดการกับนามวลี ซึ่งในตัวอย่างคือ my pen

เราสามารถเขียนกฎไวยากรณ์ให้เป็นประโยคโครงสร้างภาษาไทยได้ดังนี้

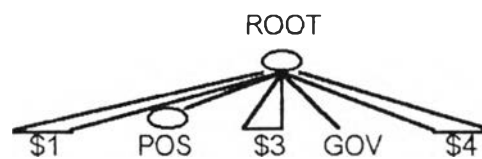
```
(GS-RULE NP-SUBD=POS
($1 @POS $3 GOV $4)
( (ROOT (& NP KLOCK))
(POS (& DPOS ^CATLOCK CAT))
(GOV (& GOV GOVLOCK))))
($1 $3 GOV $4 POS)
( (POS (:= DES CATLOCK)) ) )
```

ในกฎไวยากรณ์นี้

ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

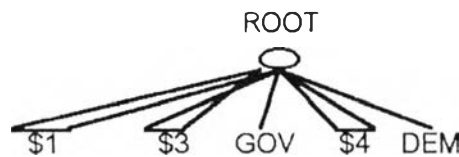
ชื่อกฎไวยากรณ์ = NP-SUBD=POS

รูปแบบต้นไม้เข้า = (\$1 @POS \$3 GOV \$4) มีรูปแบบเป็น



เงื่อนไข = ถ้านามวลีมีคำบ่งชี้ประเภท DPOS ตามด้วยคำนาม

รูปแบบต้นไม้ออก = (\$1 \$3 GOV \$4 POS) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

POS = มีหน้าที่ในประโยคคือ เป็นคำบ่งชี้ที่ขยายคำนาม (designation)

เมื่อผ่านขั้นตอนนี้ จะมีต้นไม้ย่อยของนามวลี คือ



5.2.8 .3 ส่วนขยายนามที่เป็นตัวบ่งชี้ (Determiner) ที่เหลือ คือ

- 1) คำบ่งชี้เฉพาะ เมื่อกล่าวถึงเป็นครั้งที่สอง หรือต้องการกล่าวเน้นโดยเฉพาะ (DDEF) ได้แก่ the
- 2) คำบ่งชี้ที่ไม่เฉพาะเจาะจง และมีความหมายเป็นเอกพจน์ (DINDEF) ได้แก่ a, an
- 3) คำบ่งชี้ที่ไม่ชี้เฉพาะที่ใช้แทนคนหรือสิ่งทั่ว ๆ ไป ไม่มีขอบเขตแน่นอน (DQTF) เช่น some, each, every, no
- 4) คำบ่งชี้ที่ชี้แทนคำนามที่เชื่อมประโยคสองประโยคเข้าด้วยกัน (DINT) เช่น which, whichever, whatever

ตัวอย่างประโยคเช่น

I	want	some	sugar.	
ฉัน	อยากได้	สักนิด	น้ำตาล	
ฉัน	อยากได้	น้ำตาล	สักนิด	(โครงสร้างภาษาไทย)

ในกรณีนี้เป็นการจัดการกับนามวลี ซึ่งในตัวอย่างคือ some sugar เราสามารถเขียนกฎไวยากรณ์ ให้เป็นประโยคโครงสร้างภาษาไทยได้ดังนี้

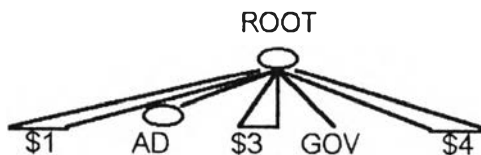
```
(GS-RULE NP-CATA+CATD
($1 @AD $3 GOV $4)
( (ROOT (& NP KLOCK))
(AD (& ^CATLOCK CAT (MEM D '(DQTF DINT DDEF DINDEF))))
(GOV (& GOV GOVLOCK)) )
($1 $3 GOV $4 AD)
( (AD (:= CATLOCK) ) ) )
```

ในกฎไวยากรณ์นี้

ชื่อชั้นตอน = GS-RULE หมายถึง ชั้นสร้างโครงสร้างประโยค (structural generation)

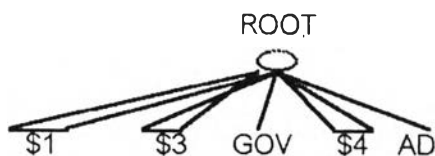
ชื่อกฎไวยากรณ์ = NP-CATA+CATD

รูปแบบต้นไม้เข้า = (\$1 @AD \$3 GOV \$4) มีรูปแบบเป็น



เงื่อนไข = ถ้านามวลีมีคำบ่งชี้ประเภท DQTF หรือ DINT หรือ DDEF หรือ DINDEF
ตามด้วยคำนาม

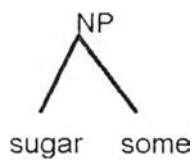
รูปแบบต้นไม้ ออก = (\$1 \$3 GOV \$4 AD) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

AD = Lock นามวลีไม่ให้ทำซ้ำ

เมื่อผ่านขั้นตอนนี้ จะมีต้นไม้ย่อยของนามวลี คือ



5.2.8.4 คำขยายเป็นคุณศัพท์ ตัวอย่างนามวลี เช่น

black cat
ดำ แมว
ที่ถูกคือ แมวดำ

เราสามารถเขียนกฎไวยากรณ์ ให้เป็นประโยคโครงสร้างภาษาไทยได้ดังนี้

(GS-RULE* NP-AP0

(\$1 (@AP \$2 AGOV \$5) \$3 GOV \$4)

((ROOT (& NP KLOCK))

(AP (& (NULL SFLOCK) AP))

(GOV (& GOV GOVLOCK))

(AGOV (& GOV ^ADJB4))

(\$3 (N== K 'AP)))

(\$1 \$3 GOV \$4 (AP \$2 AGOV \$5))

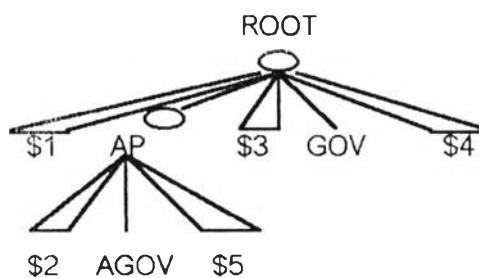
((AP (:= ATG SFLOCK KLOCK))))

ในกฎไวยากรณ์นี้

ชื่อชั้นตอน = GS-RULE หมายถึง ชั้นสร้างโครงสร้างประโยค (structural generation)

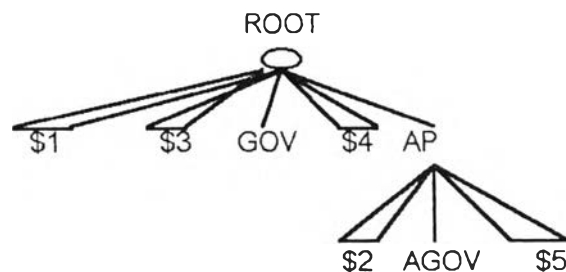
ชื่อกฎไวยากรณ์ = NP-AP0

รูปแบบต้นไม้เข้า = (\$1 (@AP \$2 AGOV \$5) \$3 GOV \$4) มีรูปแบบเป็น



เงื่อนไข = ถ้านามวลีมีคุณศัพท์วลี ตามด้วยค่านาม

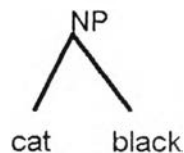
รูปแบบต้นไม้ออก = (\$1 \$3 GOV \$4 (AP \$2 AGOV \$5)) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

AP = มีหน้าที่ในประโยคคือ เป็นคำขยายของคำหลัก (attribute of governor)

เมื่อผ่านขั้นตอนนี้ จะมีต้นไม้ย่อยของนามวลี คือ



5.2.8.5 คำขยายเป็นคำนาม ตัวอย่างนามวลีเช่น

a flower garden

ดอกไม้ สวน

ที่ถูกคือ สวนดอกไม้

เราสามารถเขียนกฎไวยากรณ์ ให้เป็นประโยคโครงสร้างภาษาไทยได้ดังนี้

```

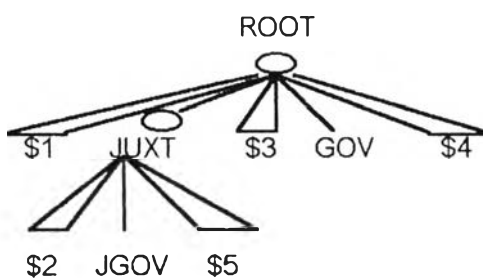
(GS-RULE* NP-JUXT=AFTER
($1 (@JUXT $2 JGOV $5) $3 GOV $4)
( (ROOT (& NP KLOCK))
  (JUXT (& JUXT ^SFLOCK))
  (JGOV GOV )
  (GOV (& GOV GOVLOCK ^PROPER)))
($1 $3 GOV $4 (JUXT $2 JGOV $5))
( (JUXT (: = COMP SFLOCK KLOCK) ) ) )
  
```

ในกฎไวยากรณ์นี้

ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

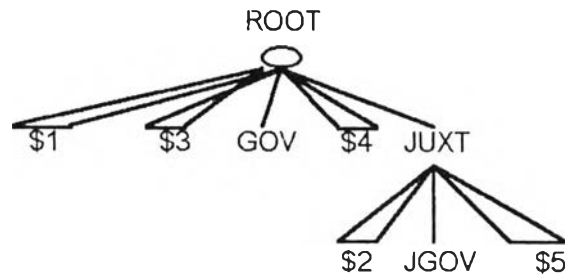
ชื่อกฎไวยากรณ์ = NP-JUXT=AFTER

รูปแบบต้นไม้เข้า = (\$1 (@JUXT \$2 JGOV \$5) \$3 GOV \$4) มีรูปแบบเป็น



เงื่อนไข = ถ้านามวลีมีคำนาม ตามด้วยคำนาม

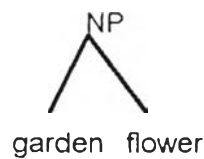
รูปแบบต้นไม้ ออก = ((\$1 \$3 GOV \$4 (JUXT \$2 JGOV \$5)) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

JUXT = มีหน้าที่ในประโยคคือ เป็น ส่วนประกอบของคำนาม (complement of noun)

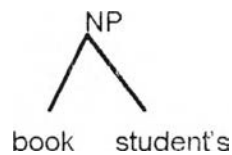
เมื่อผ่านขั้นตอนนี้ จะมีต้นไม้ย่อยของนามวลี คือ



5.2.8.6 คำขยายที่เป็นคำนามแสดงเจ้าของ ตัวอย่างนามวลี เช่น

The student's book
นักเรียน หนังสือ

นามวลีนี้ใช้กฎไวยากรณ์อันเดียวกับข้อ 5.8.2.5 คือ กฎไวยากรณ์ NP-JUXT=AFTER ก็
จะได้ต้นไม้ย่อยคือ



เราจึงเขียนเพียงกฎไวยากรณ์ที่เดิมคำว่า "ของ" เข้าไปเท่านั้น คือ

```

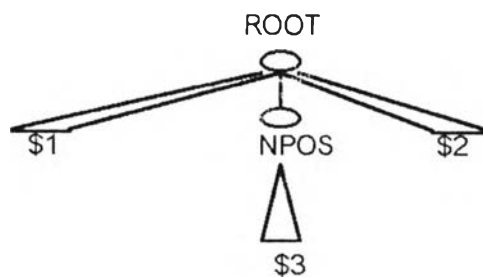
(GS-RULE* NPOS
($1 (@NPOS $3) $2)
((NPOS (& NPOS (== CAT 'N'))))
($1 POS NPOS $2)
( (POS (:= (BUL "ของ"))))
(NPOS (:= -NPOS))))
  
```

ในกฎไวยากรณ์นี้

ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

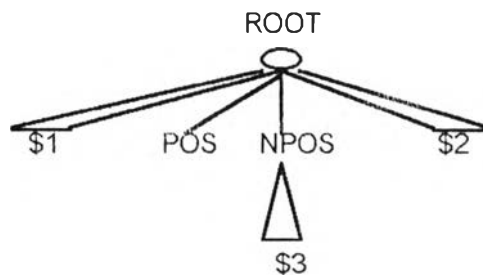
ชื่อกฎไวยากรณ์ = NPOS

รูปแบบต้นไม้เข้า = (\$1 (@NPOS \$3) \$2) มีรูปแบบเป็น



เงื่อนไข = ถ้านามวลีมีค่านามที่มีลักษณะเป็นค่านามที่แสดงความเป็นเจ้าของ

รูปแบบต้นไม้ออก = (\$1 POS NPOS \$2) มีรูปแบบเป็น

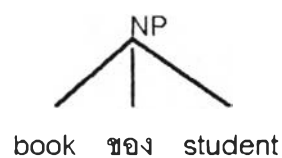


ลักษณะทางภาษาเพิ่มเติม

POS = คำแปลว่า "ของ"

NPOS = NIL

เมื่อผ่านขั้นตอนนี้ จะมีต้นไม้ย่อยของนามวลี คือ



5.2.8.7 คำขยายเป็นคำขยายนามวลี (ADV N) ตัวอย่างนามวลี เช่น

a little cat
 น้อย แมว
 ที่ถูกคือ แมวน้อย

เราสามารถเขียนกฎไวยากรณ์ให้เป็นประโยคโครงสร้างภาษาไทยได้ดังนี้

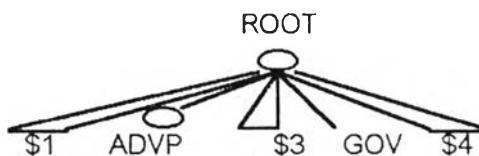
```
(GS-RULE NP-ADVNAFT=AFT
($1 @ADVNAFT $3 GOV $4)
((ROOT (& NP KLOCK))
(ADVNAFT (& ADVNAFT ADVNAFT ^CATLOCK =A))
(GOV (& GOV GOVLOCK)))
($1 $3 GOV $4 ADVNAFT)
((ADVNAFT (:= DES CATLOCK))) )
```

ในกฎไวยากรณ์นี้

ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

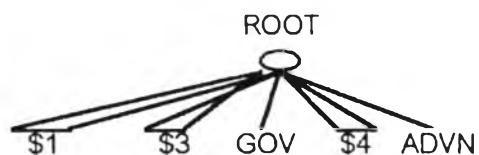
ชื่อกฎไวยากรณ์ = NP-ADVNAFT=AFT

รูปแบบต้นไม้ออก = (\$1 @ADVNAFT \$3 GOV \$4) มีรูปแบบเป็น



เงื่อนไข = ถ้านามวลีมีคำขยายนามวลี (ADVNAFT) และคำนั้นระบุในดิกชันนารีว่าเป็น ADVNAFT ตามด้วยคำนาม

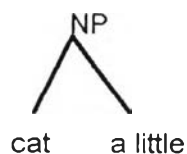
รูปแบบต้นไม้ออก = (\$1 \$3 GOV \$4 ADVNAFT) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

ADV N = มีหน้าที่ในประโยคคือ เป็นคำบ่งชี้ที่ขยายคำนาม (designation)

เมื่อผ่านขั้นตอนนี้ จะมีต้นไม้อยู่ของนามวลี คือ



5.2.9 จัดการกับประโยคปฏิเสธ โดยมีกฎไวยากรณ์ที่จัดการกับประโยคปฏิเสธดังนี้

(GS-RULE* MAI-VERB/MODAL

(\$1 @VMA \$2)

((VMA (& NEG CAT (OR V MODAL)))

(\$1 (NMEM BUL ("ก็คงจะไม่" "ไม่ได้" "จะไม่" "ไม่"))))

(\$1 MAI VMA \$2)

((VMA

(COND ((& FUT PROG PERF)

(:= ((MAI BUL) "ก็คงจะไม่"))

((OR PAST PERF)

(:= ((MAI BUL) "ไม่ได้"))

((FUT)

(:= ((MAI BUL) "จะไม่"))

(T (:= ((MAI BUL) "ไม่")))

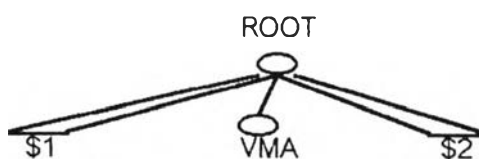
(:= -NEG -ASPEK)))

ในกฎไวยากรณ์นี้

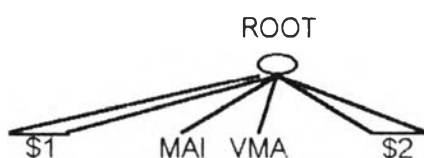
ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

ชื่อกฎไวยากรณ์ = MAI-VERB/MODAL

รูปแบบต้นไม้อัฒา = (\$1 @VMA \$2) มีรูปแบบเป็น



เงื่อนโซ่ = เป็นประโยคปฏิเสธที่มีความหมายปฏิเสธที่คำกริยาหรือกริยาช่วย
รูปแบบต้นไม้ออก = (\$1 MAI VMA \$2) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

VMA = ถ้าเป็น Future continuous perfect tense แล้ว คำแปลของ MAI คือ "ก็คงจะ

"ไม่"

ถ้าเป็น Past perfect tense แล้ว คำแปลของ MAI คือ "ไม่ได้"

ถ้าเป็น Future tense แล้ว คำแปลของ MAI คือ "จะไม่"

นอกจากนี้ แล้วคำแปลของ MAI คือ "ไม่"

5.2.10 จัดการกับกาล (Tense) ในประโยค โดยมีกฎไวยากรณ์ที่จัดการกับกาลในประโยค ดังนี้

```

(GS-RULE* PROG-OBJ1=NIL
($1 @VGOV $2)
((ROOT (MEM K '(VCL SCL RELCL PARTCL)))
(VGOV (& GOV V ^NEG (TRUE ASPEK)))
($2 ^ARG1))
($1 B4 VGOV AFT $2)
((VGOV
  ((= ASPEK 'KAMLANG)
    (:= ((B4 BUL) "กำลัง")))
  ((= ASPEK 'JA)
    (:= ((B4 BUL) "จะ")))
  ((= ASPEK 'LEEW)
    (:= ((AFT BUL) "แล้ว")))
  ((= ASPEK 'DAILEEW)
    (:= ((B4 BUL) "ได้")
      ((AFT BUL) "แล้ว"))))
(:= -ASPEK )))

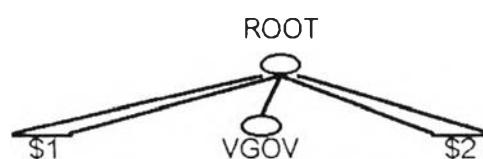
```

ในกฎไวยากรณ์นี้

ชื่อขั้นตอน = GS-RULE หมายถึง ขั้นตอนสร้างโครงสร้างประโยค (structural generation)

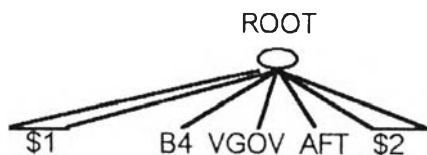
ชื่อกฎไวยากรณ์ = PROG-OBJ1=NIL

รูปแบบต้นไม้เข้า = (\$1 @VGOV \$2) มีรูปแบบเป็น



เงื่อนไข = ถ้ามีคำกริยาในประโยค และไม่เป็นประโยคปฏิเสธ

รูปแบบต้นไม้ออก = (\$1 B4 VGOV AFT \$2) มีรูปแบบเป็น



ลักษณะทางภาษาเพิ่มเติม

VGOV = ถ้า ASPEK เท่ากับ 'KAMLUNG' แล้ว คำแปลของ B4 คือ "กำลัง"

ถ้า ASPEK เท่ากับ 'JA' แล้ว คำแปลของ B4 คือ "จะ"

ถ้า ASPEK เท่ากับ 'LEEW' แล้ว คำแปลของ AFT คือ "แล้ว"

ถ้า ASPEK เท่ากับ 'DAILEEW' แล้ว คำแปลของ B4 คือ "ได้" และคำแปลของ AFT คือ "แล้ว"

6.2 การสังเคราะห์คำในประโยคภาษาไทย

การนิยามกฎไวยากรณ์สังเคราะห์คำในประโยคภาษาไทย (GM-MODEL) แตกต่างจากการนิยามกฎไวยากรณ์ที่กล่าวมาแล้วคือการเป็นนิยามโดยอาศัยโครงสร้างของต้นไม้ แต่การนิยามในส่วนนี้เรียกว่า การนิยามโดยอาศัยเงื่อนไข (conditional grammar) ซึ่งมีรูปแบบการนิยามดังนี้

```
<cond-assignment> ::= (COND (<condition> (assignment-sequence>)  

                           (<condition> (assignment-sequence>)  

                           ..... )
```

จากบทที่ 3 เมื่อเปรียบเทียบปัจจัยเปลี่ยนของภาษาอังกฤษและภาษาไทย แล้วพบว่า มีปัจจัยเปลี่ยนของภาษาอังกฤษที่สามารถแทนได้ด้วยคำภาษาไทย ดังนี้

1. ปัจจัยเปลี่ยนชนิดเปลี่ยนคำกริยาเป็นคำนามของภาษาอังกฤษ (VN) สามารถแทนได้ด้วย "การ" ในภาษาไทย เช่น develop-development-การพัฒนา
2. ปัจจัยเปลี่ยนชนิดเปลี่ยนคำคุณศัพท์เป็นคำนามของภาษาอังกฤษ (AN) สามารถแทนได้ด้วย "ความ" ในภาษาไทย เช่น dark-darkness-ความมืด
3. ปัจจัยเปลี่ยนชนิดเปลี่ยนคำคุณศัพท์เป็นคำกริยาวิเศษณ์ของภาษาอังกฤษ (AM) สามารถแทนได้ด้วย "อย่าง" ในภาษาไทย เช่น slow-slowly-อย่างช้าๆ

4. ปัจจัยเปลี่ยนชนิดเปลี่ยนคำกริยาเป็นคำนามที่เป็นบุคคลของภาษาอังกฤษ (VNAG) สามารถแทนได้ด้วย "ผู้" หรือ "นัก" ในภาษาไทย เช่น write-writer-นักเขียน
5. ปัจจัยเปลี่ยนชนิดเปลี่ยนคำนามเป็นคำนามของภาษาอังกฤษ (NN) สามารถแทนได้ด้วย "ความเป็น" ในภาษาไทย เช่น member-membership-ความเป็นสมาชิก
6. ถ้าเป็นประโยคแบบ Passive Voice ที่ไม่มีตัวกระทำ จะเพิ่มคำว่า "ถูก" เข้าไปหน้าคำกริยา

สามารถนิยามใน GM-MODEL ได้ดังนี้

```
(*GM-GRAMMAR*
((COND
(N (WHEN ^AFFIX
(IF VNAG (:= (AFFIX "ผู้")) (:= (AFFIX NAFX))))
(WHEN ^AFFIX
(IF AN (:= (AFFIX "ความ")) (:= (AFFIX NAFX))))
(WHEN ^AFFIX
(IF VN (:= (AFFIX "การ")) (:= (AFFIX NAFX))))
(WHEN ^AFFIX
(IF VINGN (:= (AFFIX "การ")) (:= (AFFIX NAFX))))
(WHEN ^AFFIX
(IF NN (:= (AFFIX "ความเป็น")) (:= (AFFIX NAFX))))
(WHEN DOLLAR (:= (AFFIX '$_')))
(WHEN (& (== BUL "เปอร์เซ็นต์") (TRUE CARD-UL))
(:= (BUL CARD-UL) (AFFIX '_%'))))

((& V ^MODAL)

(WHEN (& ^affix PASSIVE) (:= (AFFIX "ถูก")))))
```