

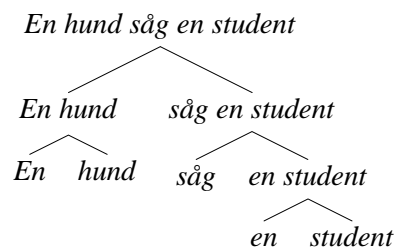
Frasstrukturgrammatik

Det här är tänkt som en ytterst elementär introduktion till frasstrukturgrammatik. En viss allmän grammatikintuition krävs för att man skall se poängen med resonemangen och kunna tillämpa dem. Avsikten med texten är att den skall introducera de formella mekanismerna från grunden.

1. Konstituentanalys

Man kan analysera meningar och andra språkliga uttryck genom att dela upp dem i delar. Om vi t.ex. har en mening som *En hund såg en student*, så kan vi kanske tänka att *En hund* är en del, ja, subjektet i traditionell satslösning, och *såg en student* den andra. Sådana delar kallas **konstituent**er. Frasen *såg en student* kan i sin tur delas upp i predikat och objekt, alltså i *såg* och *en student*. Slutligen kan flerordiga uttryck delas upp i ord.

Om vi skall visa hur vi klippt upp meningen så kan vi rita så här:

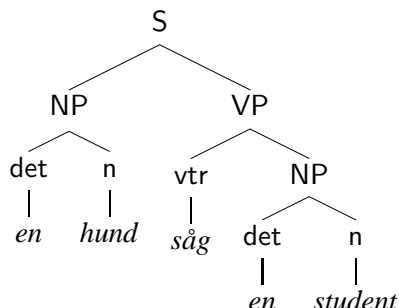


Vi kan också skriva, lite mer kompakt, $[[En\ hund]\ [såg\ [en\ student]]]$. Detta visar hur vi har klippt, och därmed vilka fraser vi urskiljt.

2. Frasstrukturträd

I frasstrukturgrammatik beskriver man meningars grammatik genom att konstruera träd som detta. Det är väsentligen samma träd som ovan, fast med ytterligare etiketter. Vi har satt in en kategorisymbol för varje ord och ersatt varje fras med en

kategorisymbol. (Vi ser ju ändå vilken fras som avses.)



I detta träd förbinds fjorton s.k. noder (knutpunkter) med streck. Dessa noder är etiketterade med symboler. Vi har tio olika symboler: S, NP, VP, det, n, vtr, *en*, *hund*, *såg*, och *student*. Fyra av dessa förekommer som etiketter på två noder, nämligen, NP, det, n, och *en*. Det finns precis en etikett på varje nod.

Jag har tänkt mig att sex av dessa skall stå för grammatiska kategorier, nämligen:

S: mening (sentence, eller tänk "sats")

NP: nominalfras

VP: verbfras

det: determinator (artikel)

n: substantiv (noun) (nomen)

vtr: transitivt verb (verb som tar ett objekt)

Dessa symboler kallas **icke-terminaler**, för trädet får inte "sluta" (terminera) i dessa. Trädet skall sluta i "riktiga" ord, som här *en*, *hund*, *såg*, och *student*. Dessa symboler kallas alltså **terminaler**, eftersom trädet slutar i dessa.

Notera att trädet är uppochnedvänt, med roten högst upp. **Rotnoden** är den som har etiketten S. Ordnoderna, eller terminalnoderna, kallas följdriktigt för **löv(noder)**.

Noderna motsvarar nu de ordsekvenser (terminalsekvenser) som man kan hitta under dem. De noder som ligger under en annan nod och som är förbundna med den kallas för denna nods **döttrar (dotternoder)**. (Och vänder vi på det hela så talar vi om en nods **moder/modernod**.) En nods döttrar visar vad ett uttryck består av. I trädet ovan så består t.ex. S-uttrycket (hela ordsekvensen) av ett NP-uttryck (*en hund*) och ett VP-uttryck (*såg en student*). Vi kan också, t.ex, se att ordet *hund* är av kategorin (ordklassen) n (substantiv).

Vi har nu tre typer av noder:

Rotnoden: Denna är den "översta" noden. Den är inte dotter till någon nod.

Icke-terminala noder: Dessa är etiketterade med icke-terminaler (grammatiska begrepp) och måste ha döttrar (som kan ta oss till terminalerna, alltså till de "riktiga" orden).

Terminala noder (löv): Dessa är etiketterade med terminaler. Eftersom vi därmed nått till de "riktiga" orden så kan vi inte ha några dotternoder till dessa.

Det finns också en linjär ordning mellan noderna, som visar den vanliga ordningen i språket, alltså den som ges av tidsföljden i tal och den vanliga läsordningen från vänster till höger när det gäller skrift.

För att vi skall ha ett träd så måste alla noder utom en (alltså rotnoden) ha precis en modernod. Vi får inte heller ha korsande grenar: Om en nod ligger före en annan, så måste den första nodens döttrar också ligga före den andra nodens döttrar.

Som vi nu konstruerat träd så är det klart att träd är formella saker, som vi uttömmande kan representera i en dator. Vi kan också utföra olika automatiska manipulationer av träd. Och vi kan också bygga träd givet regler för hur de får se ut.

3. Frässtrukturgrammatik

Som vi sett ovan, så finns det en hel del allmänna krav på syntaxträd. Men vi kan även formulera mer specifika villkor. Och skall vi rikta in oss på att bygga träd för ett visst språk, så måste vi göra det. På så sätt kan vi nu skriva *grammatiker* som är villkor på hur träd får byggas och samtidigt fångar strukturen hos det språk vi vill beskriva (t.ex. svenska). Vi kan tänka oss en grammatik som en samling av grammatiksatser, där varje grammatiksats tillåter en del av ett träd.

Nu kan vi t.ex. vilja säga vilka ord som tillhör en viss kategori. Med en sats som " $n \rightarrow \text{hund}$ " säger vi att *hund* tillhör kategorin *n* (ordklassen substantiv). Som trädvillkor innebär det att en nod med etiketten *n* får ha en nod med etiketten *hund* som enda dotter. Ja, det gällde ju precis denna del av trädet ovan:

$$\begin{array}{c} n \\ | \\ \text{hund} \end{array}$$

Sådana grammatiksatser kallas **lexikoningångar**. De säger att en visst ord (terminalsymbol) tillåts och vilken kategori (icke-terminal) det tillhör. De reglerar närmare bestämt de fall i vilka en icke-terminal nod har en (enda) terminal dotter. I en lexikoningång har vi alltså precis en icke-terminal till vänster och en terminal till höger. För trädets ovan del behövs alltså följande fyra lexikoningångar (som vi t.ex. kan skriva i bokstavsordning).

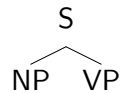
det \rightarrow en

n \rightarrow hund

n \rightarrow student

vtr \rightarrow *såg*

Vi behöver också säga i vilka fall en nod får ha (en eller fler) icke-terminala döttrar. Det gör vi med hjälp av s.k. **omskrivningsregler**. Dessa grammatiksatser säger att en nod med en given etikett får ha en sekvens av döttrar med givna etiketter. T.ex. säger grammatiksatzen "S \rightarrow NP VP" att en S-nod (alltså nod med etiketten S) får ha en NP-nod och en VP-nod som döttrar med angiven linjär ordning. Det är alltså denna del av trädet som därmed tillåts:



Vi behöver följande tre regler för att konstruera trädet i avsnitt 2:

NP \rightarrow det n

S \rightarrow NP VP

VP \rightarrow vtr NP

Ytterligare en restriktion kan vara bra att ha: Vilken symbol får stå på roten? I trädet ovan står S som rotenodens etikett. Vi kan då säga att grammatikens **startsymbol** är S, eftersom vi avsett att bygga träd för meningar.

Grammatiken bakom trädet kan nu sammanfattas på detta sätt:

Icke-terminaler: S, NP, VP, det, n, vtr.

Terminaler: *en, hund, såg, student*

Startsymbol: S.

Grammatiksatser (**produktioner** kan man också säga):

NP \rightarrow det n

S \rightarrow NP VP

VP \rightarrow vtr NP

det \rightarrow *en*

n \rightarrow *hund*

n \rightarrow *student*

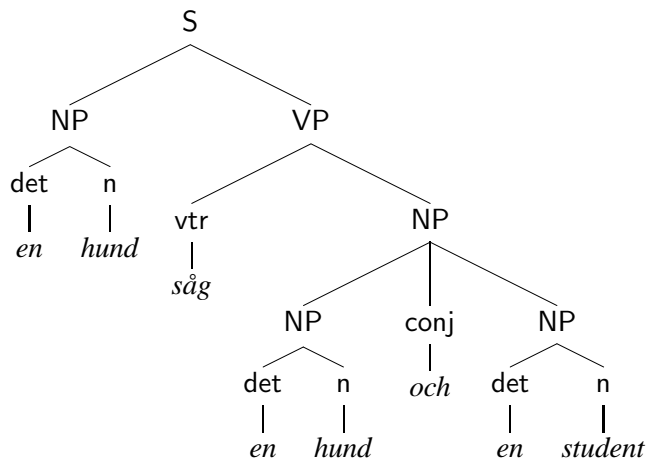
vtr \rightarrow *såg*

Denna grammatik tillåter fyra olika träd, nämligen dem för *en hund såg en hund, en hund såg en student, en student såg en hund* och *en student såg en student*.

Antalet träd växer snabbt om vi till denna grammatik lägger till nya ord: Om vi t.ex. lägger till tre nya n och tre nya vtr, så får vi $5 \times 5 \times 4 = 100$ träd, för vi väljer n på två ställen och vtr på ett (och valen är oberoende).

4.1 Exempel: samordning

Vi kan också skapa en liten grammatik som tillåter oändligt många träd. Det som kallas **Samordning** är t.ex. möjligt:

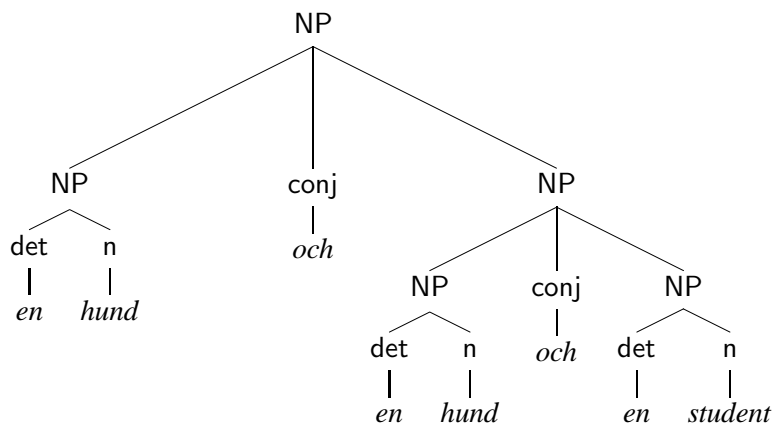


För att detta träd skall tillåtas behöver vi bara följande nya grammatiksatser:

conj \rightarrow *och*

NP \rightarrow NP conj NP

Vi inför alltså konjunktionen (conj) *och* och tillåter att två NP sätts ihop till en NP med en konjunktion (NP \rightarrow NP conj NP). Detta kallas alltså **samordning** i vanlig grammatik. Vi har bara lagt till ett ord och en regel, men det gör att vi nu kan bygga hur många och hur stora träd som helst. En NP-nod införd av den nya regeln kan i sin tur byggas ut med denna regel, som man ser i detta exempel:



(Denna NP kan t.ex. ersätta vilken som helst NP i inledningens träd.) Var och en av de NP-noder som har expanderats med regeln NP \rightarrow det n kunde lika gärna ha expanderats med NP \rightarrow NP conj NP. Det finns alltså i princip inget största träd vi kan bygga på detta sätt. Det finns alltid ett ännu större träd, och så verkar det

vara i naturligt språk också, eftersom vi vad grammatiken anbelangar alltid kan göra meningar längre och längre och mer och mer ordrika, men som tur är tröttnar vi ganska snabbt på sådan språklig produktivitet, och våra läsare och åhörare tappar antagligen tålamodet ännu snabbare.

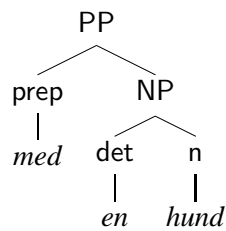
4.2 Exempel: prepositionsfraser

Prepositioner står före (det är det som termen *preposition* säger) fraser, i de allra flesta fall nominalfraser. Ett sådant par brukar kallas för en prepositionsfras (PP). Om vi gör följande tillägg till vår exempelgrammatik, så kan vi t.ex. bygga frasen *med en hund*.

prep → *med*

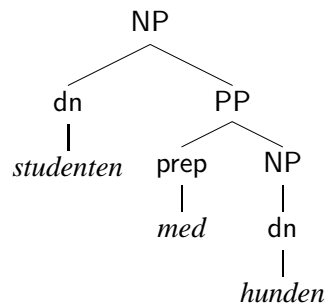
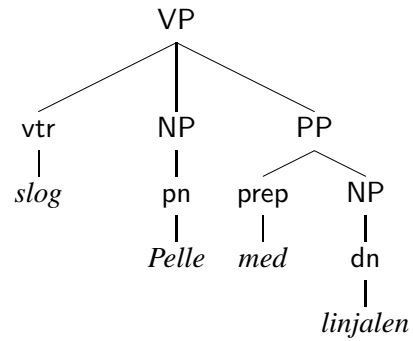
PP → prep NP

Vi får därmed bl.a. följande träd:



Nu har vi dock ingen möjlighet att sätta in en PP i ett S-träd. PP:er kan bestämma både substantiv och verb. I *studenten med hunden* knyts PP till ett substantiv, medan den hör ihop med ett verb i *slog Pelle med linjalen*.

Träden för dessa exempel borde bli något i stil med:

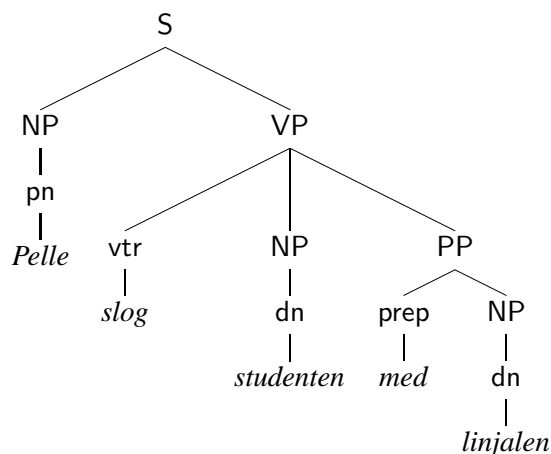
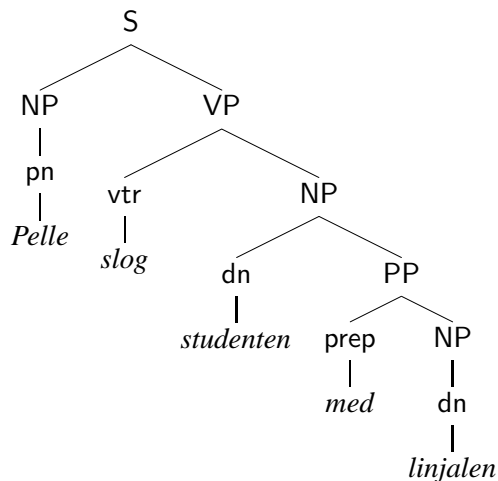


Dessa träd kräver en del nya regler och ord. Vi behöver införa några nya sorters substantiv — dn: "definite noun" (bestämt substantiv) pn: "proper noun" (egenamn). Dessa kan ensamma bilda NP:er.

- NP → dn
- NP → dn PP
- NP → pn
- VP → vtr NP PP
- dn → *hunden*
- dn → *studenten*
- dn → *linjalen*
- pn → *Pelle*

5. Grammatisk flertydighet ("ambiguitet")

Med den grammatik som vi nu infört uppstår den intressanta situationen att två olika träd kan knytas till vissa ordsekvenser:



Reglerna tillåter oss alltså att knyta PP:n som syster till både närmaste substantiv och till verbet. Om vi tänker i termer av betydelse, kan vi tänka oss att en systemnod "bestämmer" ett ord. Så *med linjalen* kan alltså säga vilken person *studenten* syftar på eller hur någon *slog* (i vilket fall prepositionen *med* antyder att det handlar om tillhygget, eller som språkvetare brukar säga, instrumentet).

Dessa två frasstrukturträd verkar alltså visa en grammatisk skillnad som svarar mot en genuin semantiskt tvetydighet.

De regler som vi infört visar också tvetydigheten hos en NP som *studenten med linjalen och hunden*. (Bra övning: Rita upp de båda möjliga träden!)

Lästips

Beskow, B., T. Lager, J. Nivre (1996) *Elementa i generativ grammatik*, Lund: Studentlitteratur.