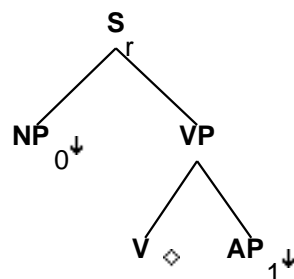


Family "Tnx0Vax1"

March 5, 2008

1 Tree "alphanx0Vax1"

1.1 graphe



1.2 comments

Adjective complement.
Simple declarative tree.

Joe felt angry
Everyone thought Joe had been feeling angry.

1.3 features

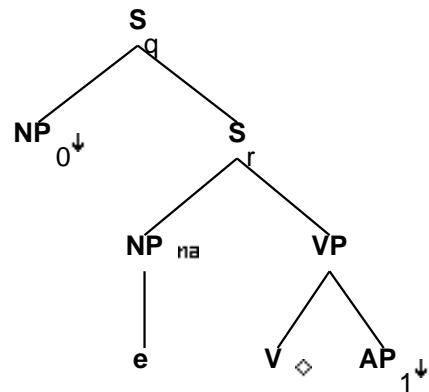
S_r.b:<extracted> = -
S_r.b:<inv> = -
S_r.b:<assign-comp> = VP.t:<assign-comp>

S_r.b:<mode> = VP.t:<mode>
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
NP₀.<agr> = S_r.b:<agr>
NP₀.<case> = S_r.b:<assign-case>
NP₀.<wh> = -
S_r.b:<wh> = NP₀.<wh>
S_r.b:<agr> = VP.t:<agr>

S_r.b:<assign-case> = VP.t:<assign-case>
 VP.b:<passive> = V.t:<passive>
 V.t:<passive> = -
 VP.b:<agr> = V.t:<agr>
 VP.b:<assign-case> = V.t:<assign-case>
 VP.b:<assign-comp> = V.t:<assign-comp>
 VP.b:<mode> = V.t:<mode>
 VP.b:<tense> = V.t:<tense>
 VP.b:<mainv> = V.t:<mainv>
 VP.b:<compar> = -
 S_r.b:<control> = NP_0.t:<control>
 S_r.b:<progressive> = VP.t:<progressive>
 S_r.b:<perfect> = VP.t:<perfect>
 S_r.b:<passive> = VP.t:<passive>
 S_r.b:<mainv> = VP.t:<mainv>

2 Tree "alphaW0nx0Vax1"

2.1 graphe



2.2 comments

Adjective complement.
 wh-extraction on the subject.

Who felt angry?
 Who did everyone think had been feeling angry?

2.3 features

S_q.b:<extracted> = +
 S_q.b:<inv> = S_r.t:<inv>
 S_q.b:<wh> = NP_0:<wh>
 S_r.t:<comp> = nil
 S_r.b:<assign-comp> = VP.t:<assign-comp>

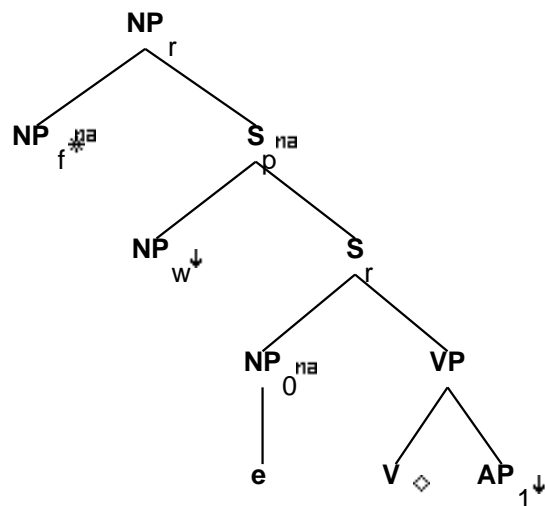
```

S_q.b:<comp> = nil
S_q.b:<mode> = S_r.t:<mode>
S_r.b:<mode> = VP.t:<mode>
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
S_r.b:<inv> = -
S_r.b:<assign-case> = NP.t:<case>
S_r.b:<agr> = NP.t:<agr>
NP.t:<trace> = NP_0.t:<trace>
NP.t:<agr> = NP_0.t:<agr>
NP.t:<case> = NP_0.t:<case>
NP.t:<wh> = NP_0.t:<wh>
NP_0:<wh> = +
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
VP.b:<passive> = V.t:<passive>
V.t:<passive> = -
VP.b:<agr> = V.t:<agr>
VP.b:<assign-case> = V.t:<assign-case>
VP.b:<assign-comp> = V.t:<assign-comp>
VP.b:<mode> = V.t:<mode>
VP.b:<tense> = V.t:<tense>
VP.b:<mainv> = V.t:<mainv>
VP.b:<compar> = -
S_r.t:<conj> = nil
S_r.b:<assign-comp> = inf_nil/ind_nil/ecm

```

3 Tree "betaN0nx0Vax1"

3.1 graphe



3.2 comments

Adjective complement.
Relative clause on the subject.

[the guy] who felt angry
[the first one] to feel angry
[the woman] that had been feeling angry

3.3 features

S_r.b:<assign-comp> = VP.t:<assign-comp>

S_r.b:<mode> = VP.t:<mode>
S_r.t:<mode> = ind/inf
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
S_r.t:<inv> = -
S_r.b:<assign-case> = NP_0.t:<case>
S_r.b:<agr> = NP_0.t:<agr>
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
VP.b:<passive> = V.t:<passive>
V.t:<passive> = -
VP.b:<agr> = V.t:<agr>

```

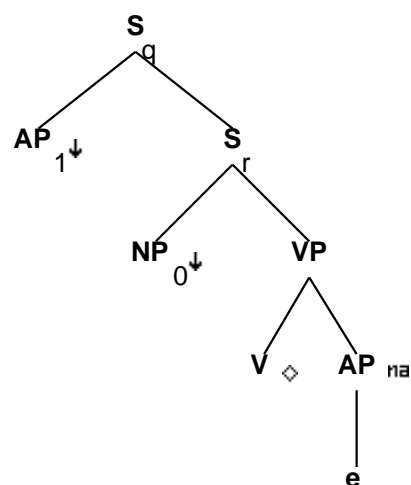
VP.b:<assign-case> = V.t:<assign-case>
VP.b:<assign-comp> = V.t:<assign-comp>
VP.b:<mode> = V.t:<mode>
VP.b:<tense> = V.t:<tense>
VP.b:<mainv> = V.t:<mainv>
VP.b:<compar> = -
S_r.t:<conj> = nil
NP_r.b:<wh> = NP_f.t:<wh>
NP_r.b:<agr> = NP_f.t:<agr>
NP_r.b:<case> = NP_f.t:<case>

NP_w.t:<trace> = NP_0.b:<trace>
NP_w.t:<case> = NP_0.b:<case>
NP_w.t:<agr> = NP_0.b:<agr>
NP_w.t:<wh> = +
S_r.t:<comp> = nil
NP_r.b:<rel-clause> = +
NP_f.b:<case> = nom/acc
NP_r.b:<pron> = NP_f.t:<pron>

```

4 Tree "alphaWA1nx0Vax1"

4.1 graphe



4.2 comments

Adjective complement.
wh-extraction of the adjective complement.

How did Joe felt?
How did everyone think Joe had felt?

4.3 features

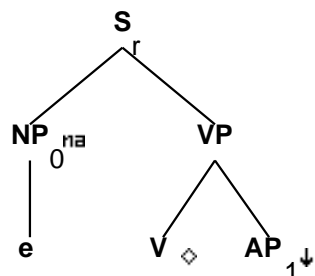
```
S_q.b:<extracted> = +

S_q.b:<inv> = S_r.t:<inv>
S_q.b:<inv> = S_q.b:<invlink>
AP_1.t:<wh> = S_q.b:<wh>
S_r.t:<comp> = nil
S_r.b:<assign-comp> = VP.t:<assign-comp>


S_q.b:<comp> = nil
S_q.b:<mode> = S_r.t:<mode>
S_r.b:<inv> = -
S_r.b:<mode> = VP.t:<mode>
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
S_r.b:<agr> = NP_0.t:<agr>
S_r.b:<assign-case> = NP_0.t:<case>
AP.t:<trace> = AP_1.t:<trace>
AP.t:<wh> = AP_1.t:<wh>
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
VP.b:<passive> = V.t:<passive>
V.t:<passive> = -
VP.b:<agr> = V.t:<agr>
VP.b:<assign-case> = V.t:<assign-case>
VP.b:<assign-comp> = V.t:<assign-comp>
VP.b:<mode> = V.t:<mode>
VP.b:<tense> = V.t:<tense>
VP.b:<mainv> = V.t:<mainv>
VP.b:<compar> = -
S_r.t:<conj> = nil
S_r.b:<control> = NP_0.t:<control>
S_r.b:<progressive> = VP.t:<progressive>
S_r.b:<perfect> = VP.t:<perfect>
S_r.b:<passive> = VP.t:<passive>
S_r.b:<mainv> = VP.t:<mainv>
```

5 Tree "alphaInx0Vax1"

5.1 graphe



5.2 comments

Adjective complement.

Imperative.

Feel angry!

Look happy.

5.3 features

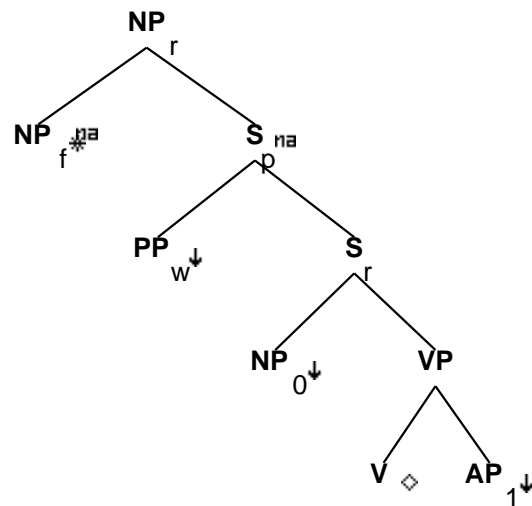
```
S_r.b:<extracted> = -  
S_r.b:<inv> = -  
S_r.b:<mode> = imp  
S_r.t:<assign-comp> = inf_nil/ind_nil  
S_r.b:<assign-comp> = VP.t:<assign-comp>  
S_r.b:<comp> = nil  
S_r.b:<tense> = VP.t:<tense>
```

```
NP_0:<agr> = S_r.b:<agr>  
NP_0:<case> = S_r.b:<assign-case>  
NP_0:<wh> = -  
NP_0:<agr pers> = 2  
NP_0:<agr 3rdsing> = -  
NP_0:<agr num> = plur/sing  
NP_0:<case> = nom  
S_r.b:<wh> = NP_0:<wh>  
S_r.b:<agr> = VP.t:<agr>  
S_r.b:<assign-case> = VP.t:<assign-case>  
VP.t:<neg> = -  
VP.t:<mode> = base  
VP.b:<mode> = V.t:<mode>  
VP.b:<passive> = V.t:<passive>  
V.t:<passive> = -
```

VP.t:<tense> = pres
 VP.b:<agr> = V.t:<agr>
 VP.b:<assign-case> = V.t:<assign-case>
 VP.b:<assign-comp> = V.t:<assign-comp>
 VP.b:<tense> = V.t:<tense>
 VP.b:<mainv> = V.t:<mainv>
 VP.b:<compar> = -
 S_r.b:<progressive> = VP.t:<progressive>
 S_r.b:<perfect> = VP.t:<perfect>
 S_r.b:<passive> = VP.t:<passive>
 S_r.b:<mainv> = VP.t:<mainv>

6 Tree "betaNpxnx0Vax1"

6.1 graphe



6.2 comments

Adjective complement.
 Simple declarative tree.

Joe felt angry
 Everyone thought Joe had been feeling angry.

6.3 features

S_r.b:<extracted> = -
 S_r.b:<inv> = -
 S_r.b:<assign-comp> = VP.t:<assign-comp>


```

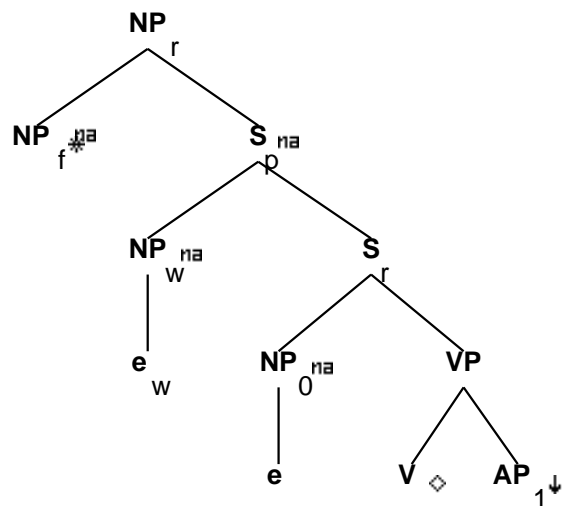
S_r.b:<mode> = VP.t:<mode>
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
NP_0:<agr> = S_r.b:<agr>
NP_0:<case> = S_r.b:<assign-case>
NP_0:<wh> = -
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
VP.b:<passive> = V.t:<passive>
V.t:<passive> = -
VP.b:<agr> = V.t:<agr>
VP.b:<assign-case> = V.t:<assign-case>
VP.b:<assign-comp> = V.t:<assign-comp>
VP.b:<mode> = V.t:<mode>
VP.b:<tense> = V.t:<tense>
VP.b:<mainv> = V.t:<mainv>
VP.b:<compar> = -
S_r.b:<control> = NP_0.t:<control>
S_r.t:<inv> = -
PP_w.t:<wh> = +
NP_r.b:<wh> = NP_f.t:<wh>
NP_r.b:<agr> = NP_f.t:<agr>
NP_r.b:<case> = NP_f.t:<case>
NP_f.b:<case> = acc/nom
S_r.t:<comp> = nil
NP_r.b:<rel-clause> = +
NP_f.b:<case> = nom/acc
NP_r.b:<pron> = NP_f.t:<pron>

S_r.b:<progressive> = VP.t:<progressive>
S_r.b:<perfect> = VP.t:<perfect>
S_r.b:<passive> = VP.t:<passive>
S_r.b:<mainv> = VP.t:<mainv>

```

7 Tree "betaNc0nx0Vax1"

7.1 graphe



7.2 comments

Relative clause of subject

[the guy] who felt angry
 [the first one] to feel angry
 [the woman] that had been feeling angry

7.3 features

S_r.b:<assign-comp> = VP.t:<assign-comp>

S_r.b:<mode> = VP.t:<mode>
 S_r.b:<comp> = nil
 S_r.b:<tense> = VP.t:<tense>
 S_r.t:<inv> = -
 S_r.b:<assign-case> = NP₀.t:<case>
 S_r.b:<agr> = NP₀.t:<agr>
 S_r.b:<agr> = VP.t:<agr>
 S_r.b:<assign-case> = VP.t:<assign-case>
 VP.b:<passive> = V.t:<passive>
 V.t:<passive> = -
 VP.b:<agr> = V.t:<agr>
 VP.b:<assign-case> = V.t:<assign-case>
 VP.b:<assign-comp> = V.t:<assign-comp>
 VP.b:<mode> = V.t:<mode>

```

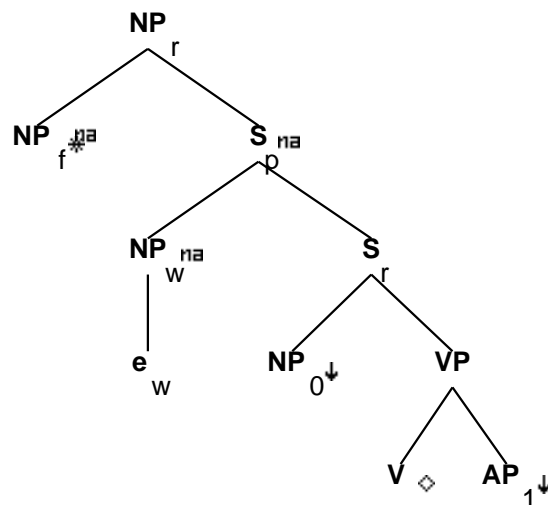
VP.b:<tense> = V.t:<tense>
VP.b:<mainv> = V.t:<mainv>
VP.b:<compar> = -
S_r.t:<conj> = nil
NP_r.b:<wh> = NP_f.t:<wh>
NP_r.b:<agr> = NP_f.t:<agr>
NP_r.b:<case> = NP_f.t:<case>

NP_w.t:<trace> = NP_0.b:<trace>
NP_w.t:<case> = NP_0.b:<case>
NP_w.t:<agr> = NP_0.b:<agr>
NP_r.b:<rel-clause> = +
S_r.t:<mode> = inf/ger/ind
S_r.t:<nocomp-mode> = inf/ger
VP.t:<assign-comp> = that/ind_nil/inf_nil/ecm
S_r.b:<nocomp-mode> = S_r.b:<mode>
NP_f.b:<case> = nom/acc
NP_r.b:<pron> = NP_f.t:<pron>

```

8 Tree "betaNcnx0Vax1"

8.1 graphe



8.2 comments

Adjective complement.
Simple declarative tree.

Joe felt angry
Everyone thought Joe had been feeling angry.

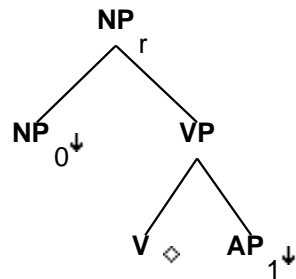
8.3 features

```
S_r.b:<extracted> = -  
S_r.b:<inv> = -  
S_r.b:<assign-comp> = VP.t:<assign-comp>
```

```
S_r.b:<mode> = VP.t:<mode>  
S_r.b:<comp> = nil  
S_r.b:<tense> = VP.t:<tense>  
NP_0.<agr> = S_r.b:<agr>  
NP_0.<case> = S_r.b:<assign-case>  
NP_0.<wh> = -  
S_r.b:<agr> = VP.t:<agr>  
S_r.b:<assign-case> = VP.t:<assign-case>  
VP.b:<passive> = V.t:<passive>  
V.t:<passive> = -  
VP.b:<agr> = V.t:<agr>  
VP.b:<assign-case> = V.t:<assign-case>  
VP.b:<assign-comp> = V.t:<assign-comp>  
VP.b:<mode> = V.t:<mode>  
VP.b:<tense> = V.t:<tense>  
VP.b:<mainv> = V.t:<mainv>  
VP.b:<compar> = -  
S_r.b:<control> = NP_0.t:<control>  
NP_r.b:<wh> = NP_f.t:<wh>  
NP_r.b:<agr> = NP_f.t:<agr>  
NP_r.b:<case> = NP_f.t:<case>  
NP_f.b:<case> = acc/nom  
S_r.t:<inv> = -  
S_r.t:<mode> = ind/inf  
S_r.t:<nocomp-mode> = ind  
VP.t:<assign-comp> = that/for/ind_nil  
S_r.b:<nocomp-mode> = S_r.b:<mode>  
NP_r.b:<rel-clause> = +  
NP_f.b:<case> = nom/acc  
NP_r.b:<pron> = NP_f.t:<pron>  
  
S_r.b:<progressive> = VP.t:<progressive>  
S_r.b:<perfect> = VP.t:<perfect>  
S_r.b:<passive> = VP.t:<passive>  
S_r.b:<mainv> = VP.t:<mainv>
```

9 Tree "alphaGnx0Vax1"

9.1 graphe



9.2 comments

Adjective complement - NP Gerund:

[Joe('s) feeling unhappy] bothered his mother.

9.3 features

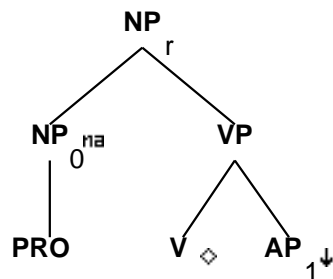
NP_r.b:<case> = nom/acc
NP_r.b:<agr num> = sing
NP_r.b:<agr pers> = 3
NP_r.b:<agr 3rdsing> = +
NP_r.b:<gerund> = +

NP_0:<wh> = NP_r.b:<wh>
VP.t:<mode> = ger

VP.b:<mode> = V.t:<mode>
VP.b:<passive> = V.t:<passive>
V.t:<passive> = -
VP.b:<compar> = -
NP_0:<case> = acc/gen

10 Tree "alphaGnx0Vax1-PRO"

10.1 graphe



10.2 comments

Adjective complement
NP gerund w/ PRO subject

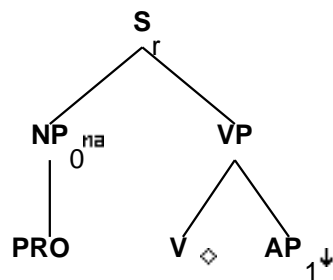
[PRO feeling unhappy] bothered Joe.

10.3 features

NP_r.b:<case> = nom/acc
NP_r.b:<agr num> = sing
NP_r.b:<agr pers> = 3
NP_r.b:<agr 3rdsing> = +
NP_r.b:<gerund> = +
NP_0:<wh> = NP_r.b:<wh>
NP_0.t:<wh> = -
NP_0.t:<case> = none
VP.t:<mode> = ger
VP.b:<mode> = V.t:<mode>
VP.b:<passive> = V.t:<passive>
V.t:<passive> = -
VP.b:<compar> = -

11 Tree "alphanx0Vax1-PRO"

11.1 graphe



11.2 comments

Adjective complement w/ PRO subject

Joe wanted [PRO to feel jolly].

While [PRO becoming angry] Joe watched his face turn red.

11.3 features

```
S_r.b:<extracted> = -
S_r.b:<inv> = -
S_r.b:<assign-comp> = VP.t:<assign-comp>
S_r.b:<mode> = VP.t:<mode>
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
NP_0:<agr> = S_r.b:<agr>
NP_0:<wh> = -
NP_0.t:<case> = none
S_r.b:<wh> = NP_0:<wh>
S_r.b:<assign-case> = NP_0.t:<case>
S_r.b:<agr> = VP.t:<agr>
VP.b:<passive> = V.t:<passive>
V.t:<passive> = -
VP.b:<agr> = V.t:<agr>
VP.b:<assign-comp> = V.t:<assign-comp>
VP.b:<mode> = V.t:<mode>
VP.b:<tense> = V.t:<tense>
VP.b:<mainv> = V.t:<mainv>
VP.b:<compar> = -
S_r.b:<control> = NP_0.t:<control>
S_r.b:<progressive> = VP.t:<progressive>
S_r.b:<perfect> = VP.t:<perfect>
S_r.b:<passive> = VP.t:<passive>
S_r.b:<mainv> = VP.t:<mainv>
VP.t:<mode> = inf/ger
```