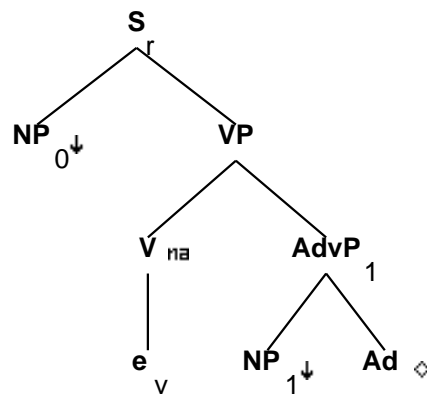


Family "Tnx0nx1ARB"

March 5, 2008

1 Tree "alphanx0nx1ARB"

1.1 graphe



1.2 comments

Declarative tree for predicative locative phrases such as 'three blocks ahead' and 'seven kilometers downstream'.

This tree family, like other predicative tree families, is anchored by the predicated object (here, the Ad), with the verb, if any, adjoining in.

EX: John is many feet ahead.

The emu is five houses down.

1.3 features

S_r.b:<extracted> = -

S_r.b:<inv> = -

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

VP.b:<compar> = -

S_r.b:<mode> = VP.t:<mode>

S_r.b:<mainv> = VP.t:<mainv>

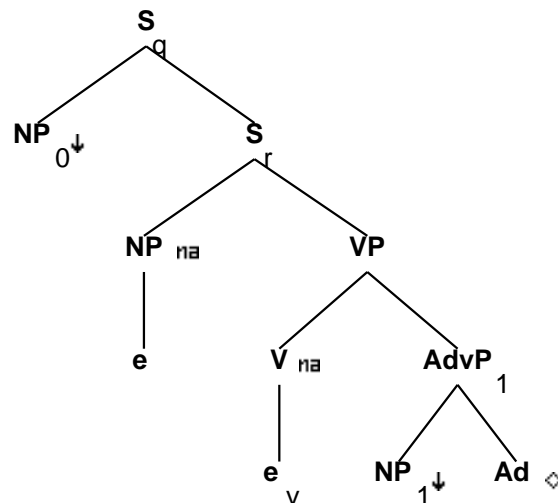
```

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_1.t:<case> = nom/acc
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<passive> = VP.t:<passive>
VP.t:<passive> = -
VP.b:<mode> = prep
VP.b:<assign-case> = acc
S_r.b:<control> = NP_0.t:<control>
AdvP_1.b:<wh> = NP_1:<wh>

```

2 Tree "alphaW0nx0nx1ARB"

2.1 graphe



2.2 comments

Wh subject extraction tree for predicative locative phrases, such as 'five meters away'. This tree does wh+ sentences only, no topicalization, since subject can not topicalize.

This tree family, like other predicative tree families, is anchored by the predicated object (here, the Ad), with the verb, if any, adjoining in.

EX: who is many kilometers away?

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>

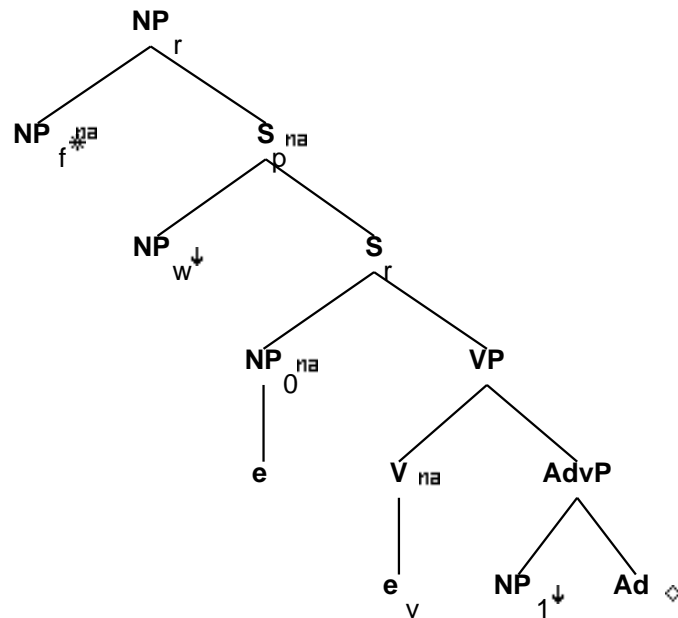
VP.b:<compar> = -
VP.t:<passive> = -
S_q.b:<comp> = nil
S_q.b:<mode> = S_r.t:<mode>
S_r.b:<mode> = VP.t:<mode>
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
S_r.b:<inv> = -
NP:<trace> = NP_0:<trace>
NP:<agr> = NP_0:<agr>
NP:<case> = NP_0:<case>
NP:<wh> = NP_0:<wh>
NP_0:<wh> = +
NP_1.t:<case> = nom/acc
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<agr> = NP.t:<agr>
S_r.b:<assign-case> = NP.t:<case>
VP.b:<mode> = prep
VP.b:<assign-case> = acc
S_r.t:<conj> = nil
S_r.b:<assign-comp> = inf_nil/ind_nil/ecm

AdvP_1.b:<wh> = NP_1:<wh>

```

3 Tree "betaN0nx0nx1ARB"

3.1 graphe



3.2 comments

Tree for predicative locative phrases with a relative clause on the subject. This tree uses a wh+ NP such as 'who' rather than a Comp.

EX: The man who is many nautical miles offshore will sink soon.

3.3 features

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

VP.b:<compar> = -
 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>
 S_r.b:<mainv> = VP.t:<mainv>
 S_r.b:<passive> = VP.t:<passive>

4 Tree "betaNc0nx0nx1ARB"

```

graph TD
    NP_r[NPr] --- NP_na1[NPna]
    NP_r --- S_na[Sna]
    NP_na1 --- f[f]
    NP_na1 --- star[*]
    S_na --- P_na[Pna]
    S_na --- S_r[Sr]
    P_na --- NP_na2[NPna]
    P_na --- S_r2[Sr]
    NP_na2 --- w[w]
    NP_na2 --- e1[e]
    S_r2 --- NP_na3[NPna]
    S_r2 --- VP_na[VPna]
    NP_na3 --- 0[0]
    NP_na3 --- e2[e]
    VP_na --- V_na[Vna]
    VP_na --- AdvP_na[AdvPna]
    V_na --- e3[e]
    AdvP_na --- NP_1[NP1]
    AdvP_na --- Ad_v[Adv]
    NP_1 --- 1[1]
    NP_1 --- v1[v]
    Ad_v --- v2[v]
    Ad_v --- v3[v]
  
```

EX: The man that is seven miles away just left.

4.3 features

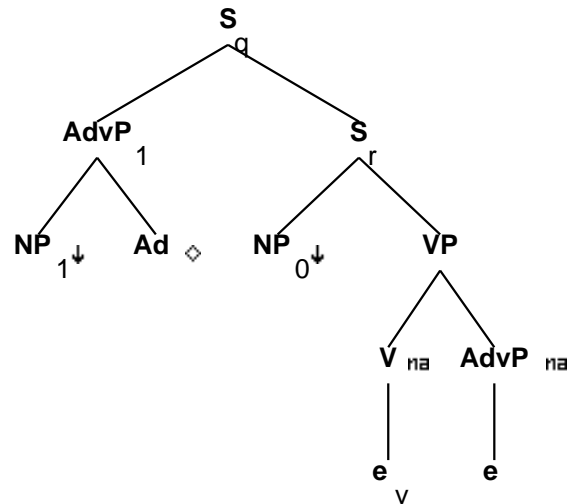
S_r.b:<assign-comp> = VP.t:<assign-comp>

VP.b:<compar> = -
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> = 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>
S_r.b:<mainv> = VP.t:<mainv>
S_r.b:<passive> = VP.t:<passive>
VP.t:<passive> = -
VP.b:<mode> = nom
VP.b:<assign-case> = acc
NP_r.b:<wh> = NP_f.t:<wh>
NP_r.b:<agr> = NP_f.t:<agr>
NP_r.b:<case> = NP_f.t:<case>
S_r.t:<conj> = nil

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/nom
S_r.t:<nocomp-mode> = inf/ger/nom
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_1.t:<case> = nom/acc
NP_r.b:<pron> = NP_f.t:<pron>

5 Tree "alphaW1ARBnx0nx1ARB"

5.1 graphe



5.2 comments

Tree for predicative locative phrase that has been wh-moved. This tree, like the declarative version, is anchored by the locative adverb in the locative phrase.

EX: How many miles ahead is John?

5.3 features

S_q.b:<extracted> = +

S_q.b:<inv> = S_r.t:<inv>

S_q.b:<inv> = S_q.b:<invlink>

S_q.b:<wh> = AdvP_1.t:<wh>

S_r.t:<comp> = nil

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

VP.b:<compar> = -

S_q.b:<mode> = S_r.t:<mode>

S_q.b:<comp> = nil

S_r.b:<mode> = VP.t:<mode>

S_r.b:<comp> = nil

S_r.b:<inv> = -

NP_0:<agr> = S_r.b:<agr>

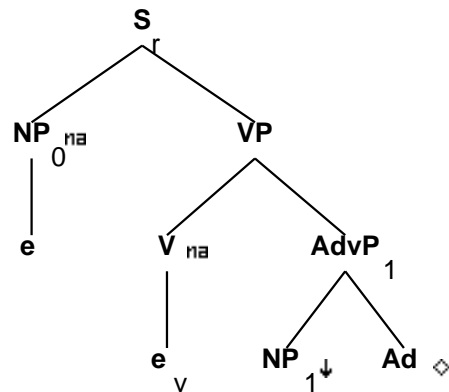
NP_0:<case> = S_r.b:<assign-case>

NP_1.t:<case> = nom/acc
 S_r.b:<agr> = VP.t:<agr>
 S_r.b:<assign-case> = VP.t:<assign-case>
 S_r.b:<control> = NP_0.t:<control>
 S_r.b:<tense> = VP.t:<tense>
 S_r.b:<mainv> = VP.t:<mainv>
 S_r.b:<passive> = VP.t:<passive>
 VP.t:<passive> = -
 AdvP:<trace> = AdvP_1:<trace>
 AdvP:<wh> = AdvP_1.t:<wh>
 AdvP_1.b:<wh> = NP_1.t:<wh>

 VP.b:<mode> = prep
 VP.b:<assign-case> = acc
 S_r.t:<conj> = nil

6 Tree "alphaInx0nx1ARB"

6.1 graphe



6.2 comments

Imperative tree for predicative locative phrases.

EX: Be three miles away!

6.3 features

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

VP.b:<compar> = -
 S_r.b:<mode> = imp

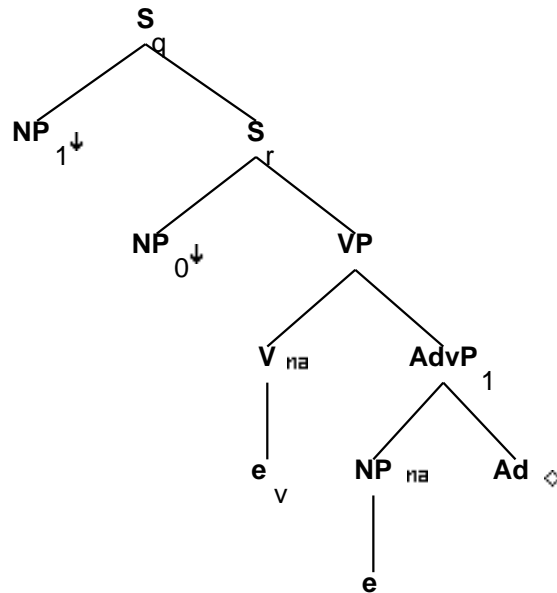

```

S_r.b:<mainv> = VP.t:<mainv>
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
NP_1.t:<case> = nom/acc
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<passive> = VP.t:<passive>
VP.t:<passive> = -
VP.t:<tense> = pres
VP.t:<mode> = base
VP.t:<neg> = -
VP.b:<mode> = prep
VP.b:<assign-case> = acc
AdvP_1.b:<wh> = NP_1:<wh>

```

7 Tree "alphaW1nx0nx1ARB"

7.1 graphe



7.2 comments

Declarative tree for predicative locative phrases such as 'three blocks ahead'

and 'seven kilometers downstream'.

This tree family, like other predicative tree families, is anchored by the predicated object (here, the Ad), with the verb, if any, adjoining in.

EX: John is many feet ahead.

The emu is five houses down.

7.3 features

S_r.b:<inv> = -

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

VP.b:<compar> = -

S_r.b:<mode> = VP.t:<mode>

S_r.b:<mainv> = VP.t:<mainv>

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>

S_r.b:<agr> = VP.t:<agr>

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

VP.b:<mode> = prep

VP.b:<assign-case> = acc

AdvP_1.b:<wh> = NP_1:<wh>

S_r.b:<control> = NP_0.t:<control>

S_q.b:<wh> = NP_1:<wh>

S_q.b:<extracted> = +

S_q.b:<inv> = S_q.b:<invlink>

S_q.b:<inv> = S_r.t:<inv>

S_q.b:<mode> = S_r.t:<mode>

S_q.b:<comp> = nil

S_r.t:<comp> = nil

S_r.t:<conj> = nil

V.t:<punct struct> = nil

NP.t:<trace> = NP_1.t:<trace>

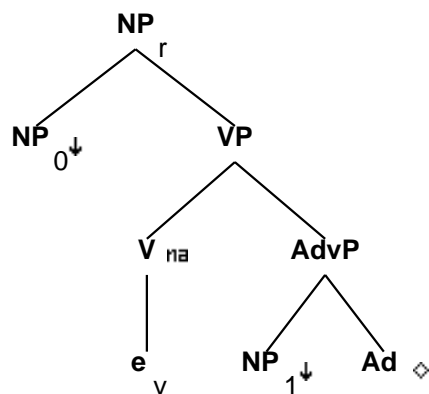
NP.t:<agr> = NP_1.t:<agr>

NP.t:<case> = NP_1.t:<case>

NP.t:<wh> = NP_1.t:<wh>

8 Tree "alphaGnx0nx1ARB"

8.1 graphe



8.2 comments

Gerund NP tree for predicative locative phrases such as 'three miles away'.

This tree family, like other predicative tree families,

is anchored by the predicated object (here, the Ad), with the verb, if any, adjoining in.

There is no corresponding D tree (*the being of three meters ahead; *the being three miles

[John('s) being seven feet downstream] is fine because there are more fish there.

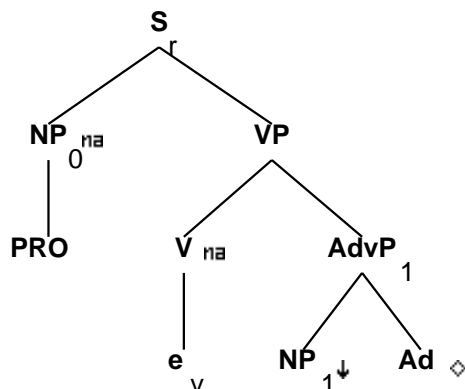
8.3 features

NP_0:<wh> = NP_r.b:<wh>
NP_1.t:<case> = nom/acc
VP.t:<mode> = ger
NP_r.b:<case> = nom/acc
NP_r.b:<agr num> = sing
NP_r.b:<agr pers> = 3
NP_r.b:<agr 3rdsing> = +
VP.b:<mode> = prep
VP.b:<assign-case> = acc
VP.b:<compar> = -

NP_r.b:<gerund> = +
AdvP_1.b:<wh> = NP_1:<wh>
NP_0:<case> = acc/gen

9 Tree "alphanx0nx1ARB-PRO"

9.1 graphe



9.2 comments

Predicative locative phrases w/ PRO subject.

This tree family, like other predicative tree families, is anchored by the predicated object (here, the Ad), with the verb, if any, adjoining in.

John wants [PRO to be many miles ahead].

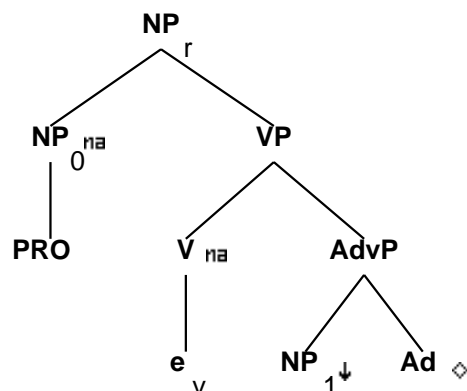
9.3 features

```

S_r.b:<extracted> = -
S_r.b:<inv> = -
S_r.b:<assign-comp> = VP.t:<assign-comp>
VP.b:<compar> = -
S_r.b:<mode> = VP.t:<mode>
S_r.b:<mainv> = VP.t:<mainv>
S_r.b:<comp> = nil
S_r.b:<tense> = VP.t:<tense>
S_r.b:<assign-case> = NP_0.t:<case>
NP_0:<agr> = S_r.b:<agr>
NP_0:<wh> = -
NP_0.t:<case> = none
NP_1.t:<case> = nom/acc
S_r.b:<agr> = VP.t:<agr>
S_r.b:<passive> = VP.t:<passive>
VP.t:<passive> = -
VP.b:<mode> = prep
VP.b:<assign-case> = acc
S_r.b:<control> = NP_0.t:<control>
AdvP_1.b:<wh> = NP_1:<wh>
VP.t:<mode> = inf/ger
  
```

10 Tree "alphaGnx0nx1ARB-PRO"

10.1 graphe



10.2 comments

Gerund NP tree for predicative locative phrases w/ PRO subject.

This tree family, like other predicative tree families, is anchored by the predicated object.

[PRO being seven feet downstream] is fine because we can still see the others.

10.3 features

NP_0:<wh> = NP_r.b:<wh>
NP_0.t:<case> = none
NP_0.t:<wh> = -
NP_1.t:<case> = nom/acc
VP.t:<mode> = ger
NP_r.b:<case> = nom/acc
NP_r.b:<agr num> = sing
NP_r.b:<agr pers> = 3
NP_r.b:<agr 3rdsing> = +
VP.b:<mode> = prep
VP.b:<assign-case> = acc
VP.b:<compar> = -
NP_r.b:<gerund> = +
AdvP_1.b:<wh> = NP_1:<wh>