The Format of Dependency Trees Output from Minipar

Dekang Lin
Department of Computer Science
University of Alberta
Edmonton, Alberta, Canada T6G 2H1
lindek@cs.ualberta.ca

June 11, 2003

A dependency structure is specified by a list of labeled tuples. Each tuple represent a word or an empty category in the sentence. The format of a labeled tuple is as follows:

```
label (word root pos governor rel extra-info extra-info ...)
```

where

`label` is a label assigned to the tuple. If the tuple represents a word in the sentence, `label` should be the index of the word in the sentence. If the tuple represents an empty category, the label can be any non-numerical string. The label for a tuple must be unique in a dependency structure.

`word` is a word in the input sentence. If the tuple represents an empty category, `word` is denoted by `()`.

`pos` is the part of speech of `word`.

`root` is the root form of `word`. If `root` is ~, `root` is the same as `word`.

`governor` is the label of the governor of `word` (if it has one). If `governor` is `*`, the tuple is the root node of the dependency tree or a fragment of a dependency tree.

`rel` is the type of the dependency relationship.

`extra-info` is a piece of additional information. Each `extra-info` is a list, with the first element in the list specifies the type of information and the remaining elements in the list providing the details. They can be used to specify, among other things, the antecedent of a word and attribute values of the word.

The following is a list of example dependency structures:

Example 1: John loves Mary

```
(
E0      (()     ~ C     *)
1       (John   _PER N  2       s       (gov love)      (ref E2))
2       (loves  love V_N_N      E0      i       (gov ~))
E2      (()     _PER N  2       subj    (gov love)      (ante 1))
3       (Mary   _PER N  2       obj     (gov love))
)
```

Line `E0` is the root of the sentence.

Line `1` represents the word "John" in the sentence.

- `_PER` is the normalized root form of all proper names classified as persons.
- `N` is the grammatical category (part-of-speech) of "John".
- `2` means that the governor of "John" is the node specified on Line `2`.
- `s` is the grammatical relationship between "John" and its governor. More specifically `s` stands for "surface subject".
- `(gov love)` means that the root form of the governor is `love`. This information is redundant since we already know that the governor of the word is on line `2`. However, by providing this information on the line, it is easier to extract word-governor relationships.
- `(ref E2)` means that "John" is co-indexed with the node on line `E2`. In this case, `E2` is the trace of "John".

Line `2` represents the word "loves".

- `love` is the root form of "loves"
- `V_N_N` is the grammatical category of "love". This symbol encodes the subcategorization frame of the verb, which list all arguments of the verb (not just objects). The first `N` in `V_N_N` means that the subject of the verb is a noun phrase. The second `N` means that the object of the verb is a noun phrase.
- `E0` means that the governor of the verb is the head of the clause on line `E0`.
- `i` is the lable assigned to relationships between the main verb of a clause and the (empty) head of the clause.
- `(gov ~)` means that the root form of the governor of "loves" is an empty string (represented by ~).

Line `E2` represents the trace of "John". Minipar assumes that the deep word order is VSO (verb-subject-object). In English, subjects are moved from their verb-internal positions to the beginning of the sentence.

- `subj` is used to lable the the relationships between "deep" subjects and their verb governors.
- `(ante 1)` means that the antecedent of `E2` is the node on Line `1`.

Line `3` represents the word "Mary".

Example 2: John is fired because he was late

```
(
E1      (()     ~ C     *)
1       (John   _PER N  3       s       (gov fire)      (ref E3))
2       (is     be be   3       be      (gov fire))
3       (fired  fire V_N_N      E1      i       (gov ~))
E3      (()     _PER N  3       obj     (gov fire)      (ante 1))
4       (because        ~ SentAdjunct   3       mod     (gov fire))
E0      (()     ~ C     4       comp1   (gov because))
5       (he     ~ N     6       s       (gov be)        (ref E4))
6       (was    be VBE  E0      i       (gov ~))
E4      (()     he N    6       subj    (gov be)        (ante 5))
7       (late   ~ A     6       pred    (gov be))
)
```

In this example, "John" is the surface subject `s` of "fire". Its trace `E3` is the object (`obj`) of "fire". "John" and its trace are linked by `(ref E3)` on line `1` and `(ante 1)` on line `E3`. The grammatical category of "because" is sentential adjunct (`SentAdjunct`). It takes a clause as argument. Like the main clause in the sentence, the complement clause of "because" is also headed by an empty category.

Example 3: who do you think will fix the car

```
(
E1      (()     ~ C     *)
1       (who    ~ N     E1      whn     (gov ~) (ref E4))
2       (do     ~ Aux   4       inv-aux (gov think))
3       (you    ~ N     4       s       (gov think)     (ref E3))
4       (think  ~ V_N_C E1      i       (gov ~))
E3      (()     you N   4       subj    (gov think)     (ante 3))
E0      (()     ~ C     4       fc      (gov think))
5       (will   ~ Aux   6       aux     (gov fix))
6       (fix    ~ V_N_N E0      i       (gov ~))
E4      (()     who N   6       subj    (gov fix)       (ante 1))
7       (the    ~ Det   8       det     (gov car))
8       (car    ~ N     6       obj     (gov fix))
)
```

- This example involves a long distance movement. The trace of `who` is the empty category on line `E4`.
- The grammatical category of `think` is `V_N_C`, which means that the subject of `think` is a noun and the object is a clause.

Example 4: She said the organization is widely recognized as the official recorder of the competitive and elusive accomplishments set in a variety of aviation adventures, in everything from hot-air balloons to Lear Jets to tiny two-seaters

```
(
E1      (()     ~ C     *)
1       (She    ~ N     2       s       (gov say)       (ref E3))
2       (said   say V_N_C       E1      i       (gov ~))
E3      (()     she N   2       subj    (gov say)       (ante 1))
E0      (()     ~ C     2       fc      (gov say))
```

```
3       (the    ~ Det   4       det     (gov organization))
4       (organization   ~ N     7       s       (gov recognize) (ref E4))
5       (is     be be   7       be      (gov recognize))
6       (widely ~ A     7       amod    (gov recognize))
7       (recognized     recognize V_N_N E0      i       (gov ~))
E4      (()     organization N  7       obj     (gov recognize) (ante 4))
8       (as     ~ Prep  7       mod     (gov recognize))
9       (the    ~ Det   10      det     (gov official))
10      (official       ~ N     8       pcomp-n (gov as))
11      (recorder       record A        10      pnmod   (gov official))
12      (of     ~ Prep  11      mod     (gov record))
13      (the    ~ Det   18      det     (gov set))
14      (competitive    ~ A     18      mod     (gov set))
15      (and)
16      (elusive        ~ A     14      conj    (gov competitive))
17      (accomplishments        accomplishment N        18      nn      (gov set))
18      (set    ~ N     12      pcomp-n (gov of))
19      (in     ~ Prep  18      mod     (gov set))
20      (a      ~ Det   21      det     (gov variety))
21      (variety        ~ N     19      pcomp-n (gov in))
22      (of     ~ Prep  21      mod     (gov variety))
23      (aviation       ~ N     24      nn      (gov adventure))
24      (adventures     adventure N     22      pcomp-n (gov of))
25      (,)
26      (in     ~ Prep  7       mod     (gov recognize))
27      (everything     ~ N     26      pcomp-n (gov in))
28      (from   ~ Prep  27      mod     (gov everything))
29      (hot-air        ~ N     30      nn      (gov balloon))
30      (balloons       balloon N       28      pcomp-n (gov from))
31      (to     ~ Prep  30      mod     (gov balloon))
32      (Lear   ~ U     33      lex-mod (gov "Lear Jets"))
33      (Jets   "Lear Jets" N   31      pcomp-n (gov to))
34      (to     ~ Prep  33      mod     (gov "Lear Jets"))
35      (tiny   ~ A     36      mod     (gov two-seater))
36      (two-seaters    two-seater N    34      pcomp-n (gov to))
)
```

Rel	Explanation	Example
amod	postverbial modifier	crashed **consistently**
appo	appositive of a noun	the CEO, **John**
aux	an auxiliary verb and main verb	**must** resign
be	'be' (used as an auxiliary verb) and main verb	**is** sleeping
being	'being' (used as an auxiliary verb) and main verb	**being** questioned
c	complementizer of a clause	**that** John likes Mary
cn	clauses used as a noun	**whoever finishes first** wins the prize
det	a determiner and its head noun	**the** dog
gen	a genitive modifier and head noun	**John's** dog
have	'have' (used as an auxiliary verb) and main verb	**have** disappeared
i	predicate and the empty head of a clause	
inv-aux	auxiary verb in a question	who **did** you win
inv-be	auxiary 'be' in a question	who **are** they talking about
inv-have	auxiary 'have' in a question	**have** you got the book
mod	an adjunct modifier and its head	**tiny** hole
nn	a prenominal modifier and its head noun	**station** manager
p-spec	the specifier of a preposiitonal phrase	
pcomp-c	clausal complement of a preposition	in **working with them**
pcomp-n	nominal complement of a preposition	in the **garden**
pnmod	postnominal adjective modifier and its head noun	files **compatible** with Excel
post	a postdeterminer and its head noun	
pre	a predeterminer and its head noun	
pred	the predicate of a clause	
rel	the head of a relative clause and its head noun	the man **who escaped**
s	surface subject of a cluase	
subj	logical subject of a verb	**John** disappeared
vrel	postnominal passive verb phrase and its head noun	the man **arrested** for **...**
wha	adverbial wh-phrase	**where** she studied
whn	nominal wh-phrase	**who** escaped
whp	prepositional wh-phrase	**in which** we trust