BNC Dependency Bank 1.0 & 2.0

Additional Documentation to Lehmann & Schneider 2012

http://www.helsinki.fi/varieng/journal/volumes/12/lehmann_schneider/

The Pro3Gres Annotation scheme

Gerold Schneider, University of Zurich

1 The Pro3Gres Annotation Scheme

The Pro3Gres parser\(^1\) automatically adds syntactic annotation to texts that are tagged and chunked. It is described in detail in Schneider (2008). The tagset required in the input texts is the Penn Treebank tagset. The syntactic representation of the parser’s output is a dependency grammar annotation. An overview of important relations is given in figure 1.

<table>
<thead>
<tr>
<th>RELATION</th>
<th>LABEL</th>
<th>EXAMPLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>verb–subject</td>
<td>subj</td>
<td>he sleeps</td>
</tr>
<tr>
<td>verb–direct object</td>
<td>obj</td>
<td>sees it</td>
</tr>
<tr>
<td>verb–second object</td>
<td>obj2</td>
<td>gave (her) kisses</td>
</tr>
<tr>
<td>verb–adjunct</td>
<td>adj</td>
<td>ate yesterday</td>
</tr>
<tr>
<td>verb–subord. clause</td>
<td>sentobj</td>
<td>saw (they) came</td>
</tr>
<tr>
<td>verb–pred. adjective</td>
<td>predadj</td>
<td>is ready</td>
</tr>
<tr>
<td>verb–prep. phrase</td>
<td>obj</td>
<td>slept in bed</td>
</tr>
<tr>
<td>noun–prep. phrase</td>
<td>modpp</td>
<td>draft of paper</td>
</tr>
<tr>
<td>noun–participle</td>
<td>modpart</td>
<td>report written</td>
</tr>
<tr>
<td>verb–complementizer</td>
<td>compl</td>
<td>to eat apples</td>
</tr>
<tr>
<td>noun–preposition</td>
<td>prep</td>
<td>to the house</td>
</tr>
</tbody>
</table>

Figure 1. Important dependency relations of Pro3Gres

The dependency grammar annotation is similar to GREVAL (Carroll et al. 2003) and to the Stanford Dependency scheme. Haverinen et al. (2008) have implemented a mapping of the Pro3Gres annotation scheme to Stanford scheme. We illustrate our set of dependency relations and give linguistic examples in this section.

Dependency Grammar (Tesnière 1959) focuses on grammatical relations, on government, on valency relations (e.g. Helbig 1992) and thus expresses the verb’s central valency relations like subject and object. The valency concept was increasingly extended to other word classes than verb (e.g. relational nouns or adjectives have valencies) and also to adjuncts in addition to complements. These relations are what we refer to as major relation types in the following. The interpretation and the linguistic motivation of these dependency labels is straightforward.

\(^1\) PR0babilistic R0bst PR0log-implemented Grammatical Relation Extraction System
According to Tesnière, dependency relations are between nuclei. The concept of nucleus roughly corresponds to noun and verb chunks (e.g. Abney). Accordingly, the dependency relations that we produce are between noun and verb chunk heads.

In order to obtain fully connected parses, every nucleus in a sentence needs to be connected. For this technical reason, also words like complementizers, prepositions and conjunctions thus need to get attached. It is not straightforward how they should be attached in a linguistically meaningful way, and many different approaches have been suggested. We attach complementizers to the subordinate verb, and prepositions to the nouns in the PP. While these decisions can be as arbitrary, they mirror the original Dependency Grammar view that content words should be used as governors. We also include them in our discussion of minor relations.

While Dependency Grammar does typically not know empty categories, long-distance dependencies often create additional, so-called secondary or indirect dependencies. For example, in the have-to-modal construction figure 2, borrower is both the overt subject of have and the implicit subject of the subordinate clause that is headed by the verb pay.

Figure 2. Long-distance dependency with a semi-modal verb

1.1 Major Relation Types

The interpretation of major relation types is typically straightforward. We illustrate the types and give examples in the following.

1.1.1 Subject

The subject relation subj expresses just what it means. Relatively many subjects occur in long-distance dependencies, which we have illustrated in figure 2. The Subject relation reports both active and passive subjects. Passive subjects alone can be searched using the 'Passive Subject' relation, which is a subset all subject relations. Results of a query for eat are shown in figure 3, an example sentence from the results in figure 4.

Figure 3. Results of a query for 'eat' with passive subjects

1.1.2 Objects: Object and Second Object

Also the object dependency is straightforward to interpret, although there are two complications. First, the complement of copular verbs like be is given an object label. Since copular verbs and regular verbs are almost always in complementary distribution, this does hardly create ambiguity. Second, there are two object relations: obj and obj2. obj is the object closer to the verb, obj2 the more remote object, if there is any. In other words: Transitive verbs except for ditransitives just have an obj relation, ditransitive verbs which have two realized objects have an obj and an obj2 relation, in which the obj is the indirect and obj2 the direct object. The example query in figure 2 finds all ditransitive verb occurrences of give found by the parser. In detail, the query works as follows. All the example we give are from the written part of the BNC, corresponding to the selection of BNC-W written in the 'Select Corpus' field. In the top filed, in line 1) we restrict the 'Head' lemma (the terms head and governor are typically used interchangeably) to give and the 'Relation' drop-down list to object. We leave the 'Dependent', i.e. the object lemma unrestricted, which means that any object will be returned, no matter what its lemma may be. We take all directions and direct or indirect link types, which are the default settings. Then we click on the more-button to formulate the query for second object. We select Second Object from the drop-down list in line 2) an place no further restrictions. Line 1 and line 2 each contain a query for a dependency relation. The 'Binding' drop-down list shown in figure 5 allows us to combine

Figure 4. Syntactic analysis of a result from figure 3.
these otherwise unrelated queries into a query for a partial tree. We leave ‘(Head1) = Head2’ which means that the head of line and line 2 are identical, or in other words, that the object and the second object attach at the same place, forming a ‘flat’ tree. If we had chosen ‘Dependent1 = Head2’ we would search for a nested tree, in this case an object that has a second object attached to it – a linguistically very unexpected structure which reports no hits.

Figure 5. Binding of relations in the query

The parse tree of an example result is given in figure 6.

1.1.3 Adjunct

The adjunct relation adj attaches NPs that are used adverbially, such as temporal expressions. An example sentence is given in figure 7.

Figure 7. Example sentence containing the adjunct relation adj

1.1.4 Prepositional Phrases

Prepositional phrases are either attached to verbs, with the relation pobj, or to nouns, with the relation modpp. A subset of the verb-attached PPs are annotated as PP-complements, with the relation pp:obj. For example, the query in figure 8 searches for PPs attached to compare. This query e.g. allows an investigation of compare with and compare to. Not that the distance between swap and the for-PP can be considerable, which poses serious problems if one does not have syntactically annotated data, and e.g. uses surface searches like windows-based approaches or regular expression queries.

Figure 8. Query for PPs attached to the verb ‘compare’

Figure 9 shows the results of a query for swap with a for-PP.

The syntactic analysis of a sentence in the result set is given in figure 10. There is a small display issue: the labels expressing the verbal attachment of the PP and additionally giving it complement status, are not clearly legible, because the labels pobj and pp:obj cover each other.

Figure 9. Results of a query for swap with a for-PP.
Finally, we give an example showing what people and governments are doing when they are under pressure. The query is shown in figure 11. In order to include only examples with high salience of under pressure, only fronted PPs (direction to the left in line 1) are included. The preposition is a dependent of the noun in the PP. The query is in line 2. This time we are searching for nested tree, in which the dependent of pressure is the preposition under. An excerpt of the results is given in figure 12.
1.1.7 Relative Clauses: modrel and modpart

Full relative clauses have the label modrel, reduced relative clauses the label modpart. In the query in figure 17 we look for nouns that are modified by the reduced relative participle extrapolate. Results are in figure 18.

Figure 16. Results for the query in figure 15

1.1.8 Predicative Adjective

The predicative adjective relation predadj attaches predicative adjectives to copular verbs. An example is given in figure 19.

Figure 19. Syntactic analysis of a sentence containing a predadj relation

1.2 Additional Relation Types

1.2.1 Preposition

The Preposition relation prep allows one to constrain a query to certain prepositions in a PP. We have seen an example query, combining a restriction to the PP starting with under with PP direction to the left, in figures 11 and 12. The syntactic analysis of one sentence in the result set is given in figure 20.

Figure 20. Analysis of a sentence containing prep with 'under' as dependent

1.2.2 Complementizer

Complementizers attach to their subordinate verb head with the compl relation. The query of figure 21 reports all instances in which it is a complementizer of a matrix clause headed by suppose. The results are in figure 22. Intuitively one might think that suppose leads to the complementizer that in most cases – here are some counterexamples. Note that the distance between suppose and if is often so long that surface searches would deliver considerable amounts of garbage.
1.2.3 Chunk-Internal Modification

According to Tesnière, dependencies are only between nuclei, which are headed by a content word. We express this by using chunks, and dependencies are only between chunk heads. Still, it is often useful to express chunk-internal relations, for example if the head is premodified by an adjective or a noun. This is exactly what the chunk-internal modification relation expresses. We have seen an example query, combining with clause subordination, in figures 15 and 16. The syntactic analysis of one sentence in the result set is given in figure 23. In order to keep the size of the syntax trees manageable for browsing, chunk-internal relations are not displayed, in our example there is no dependency link displayed between decision and the premodifying noun budget.

Figure 23. Sentence containing chunk-internal relations

1.2.4 Determiner

The Determiner dependency is always chunk-internal. For example, the word consideration appears 1639 times in subject position (according to one of our runs, depending on parser development, your count may vary a bit). 42.8% of them, 701, contain a determiner relation. Consideration also appears 2051 times in a verbally attached PP. Only 23.2% of them, 476, contain a determiner. This is partly due to the fact that the idiom take into consideration, which occurs 233 times, in all cases without a determiner. But even after deducting these counts, there is still a strong preference for determiner omission in verb-attached PPs.

1.2.5 Auxiliary

In most cases, both the auxiliary and the main verb are inside the verb chunk (as in 'have eaten' in figure 24). In questions, however, the auxiliary supporting verb gets fronted to the V2 position, and it is attached to the main verb by the auxiliary dependency. The chunk-internal and chunk-external auxiliary dependency can be queried separately or together. An example of a chunk-external syntactic analysis is given in figure 24 at the beginning of the sentence.

Figure 24. A syntactic analysis containing a chunk-external auxiliary relation

1.2.6 Adverb

The adverb relation attaches adverbs to verbs or nouns. As this relation is not based on a statistical model, its performance is low. Results for a query on adverbs of the verb eat are given in figure 25.
1.2.7 Adjective Translations
The adjective translation relation is a minor relation that is needed to allow gerunds to act as adjectives. An example is given in figure 26. The performance of this relation is quite low.

![Figure 26. Sentence containing a correctly identified adjective translation](image)

1.2.8 Noun Translation
The noun translation relation allows lonely adjectives to act as NP heads. An example is given in figure 27. The performance of this relation is very poor, so it will be reworked in the future.

![Figure 27. A noun translation](image)

1.2.9 Comparison
The comparison relation is used for comparative structures, for example the one given in figure 28.

![Figure 28. Sentence containing a comparison structure](image)

1.2.10 Conjunction
The conjunction relation is a helper relation that is used to connect conjoined phrases and clauses.

1.2.11 Apposition
The apposition relation expresses nominal and adjectival appositions. As this relation is not present in the Penn Treebank which was used for training the parser, its performance is relatively low.

1.2.12 Nominal Chunk underchunking
The chunker that is used in the preprocessing step often creates chunks that are too small. We have managed to correct some cases of underchunking, by using a relation with the label nchunk. This is a heuristic extension, the performance is relatively low and the linguistic interpretation is not always clear, but typically coincides with the chunk-internal modification. In the query tool, separate queries for nchunk are thus not available.

1.2.13 Bridge
The bridge relation is strictly speaking not a dependency relation, it connects partial parses which arise from grammar incompleteness, high parsing complexity in long sentences, fragments that are linguistically difficult to connect or hesitations and restarts in spoken language. In the majority of cases, the bridge relation rather expresses a shortcoming of the parser than a linguistically meaningful relation. In some cases, it indicates false starts or grammatical errors.
References


