Chapter 2
Predicate Argument Structures for Information Extraction from Dependency Representations: Null Elements are Missing

Rodolfo Delmonte

Abstract State of the art parsers are currently trained on converted versions of Penn Treebank into dependency representations which however don't include null elements. This is done to facilitate structural learning and prevent the probabilistic engine to postulate the existence of deprecated null elements everywhere (see [15]). However it is a fact that in this way, the semantics of the representation used and produced on runtime is inconsistent and will reduce dramatically its usefulness in real life applications like Information Extraction, Q/A and other semantically driven fields by hampering the mapping of a complete logical form. What systems have come up with are "Quasi"-logical forms or partial logical forms mapped directly from the surface representation in dependency structure. We show the most common problems derived from the conversion and then describe an algorithm that we have implemented to apply to our converted Italian Treebank, that can be used on any CONLL-style treebank or representation to produce an "almost complete" semantically consistent dependency treebank.

Keywords Predicate argument structures · Dependency structures · Null elements · Logical form · Information extraction for question answering and text understanding

1 Introduction

I take the task of Information Filtering to be in essence comparable to finding the best way to evaluate a ranking of candidates for knowing "Who" did "What", possibly "When" and "Where" and maybe sometimes also "How". Now this is what is also usually referred to as answering factoid questions. In our case also the "Did" is

R. Delmonte (✉)
Department of Linguistic Studies and Comparative Cultures and Department of Computer science, Ca' Foscari University,Venice,∼Italy
e-mail: delmont@unive.it

C. Lai et al. (eds.), *Distributed Systems and Applications of Information Filtering and Retrieval*, Studies in Computational Intelligence 515,
DOI: 10.1007/978-3-642-40621-8_2, © Springer-Verlag Berlin Heidelberg 2014

important, i.e. also function or stop words and relations are relevant, and not only "entities" or Named Entities that can be collected from the Semantic Web. More on this below. Nobody nowdays can think of doing away with NLP tools, which even in a field like Machine Translation are becoming essential to improve performance. State of the art systems are using more and more dependency representations which have lately shown great resiliency, robustness, scalability and great adaptability for semantic enrichment and processing. However, by far the majority of systems available off the shelf don't support a fully semantically consistent representation and lack Empty or Null Elements.

In his paper on the upgraded version of PennTreebank (hence PT), Marcus [22] refers explicitely to Predicate-Argument Structures (hence PASs) and to the need to address this level of annotation with the new syntactic annotation scheme proposed for the new version of PT. He mentions explicitely that "we intend to automatically extract a bank of PASs intended at the very least for parser evaluation from the resulting annotated corpus" and further on "the notation should make it easy to automatically recover PAS". He has made clear statements about the need to allow for a clear and concise distinction between verb ARGUMENTs and ADJUNCTs, however, only where this distinction is clear, that is in simple cases. In fact in the paper he then asserts that it is very difficult to make this distinction consistently. This happens to be true: the final version of PT II does not include coindexing in controversial cases and has coindexing for null SBJ only in a percentage of the cases. Over 36,862 cases of null elements (including traces, expletives, gapping and ambiguity) as listed in [18], 8,416 are not coindexed, that is 22.83 %. If we exclude all traces of WH and topicalization and limit ourselves to the category OTHER TRACES which includes all unexpressed SBJ of infinitivals and gerundives, we come up with 12,172 cases of Null non-coindexed elements, 33 % of all cases. However, this could represent a small percentage when compared to the number of null elements in languages like Chinese or Romance languages like Italian which allow for free null subjects insertion in tensed clauses. More on this topic below.

So eventually, the question is not to clutter the converted PT with information which will result to be harmful if not just useless for training purposes, since null elements don't show up in surface text. Even though the idea behind Johansson et al. effort was that of deriving a more "semantically useful" representation, we are still far apart from the need to reflect a more complex linguistically deep representation. Current statistically dependency parsers have made improvements in enriching their structural output representation [4, 7, 14, 23]. However, coindexation is not always performed: when it is, its performance is computed separately because it is lower than accuracy for labeled/unlabeled tasks. In particular, Schmid report 84 % F-score for empty elements prediction and 77 % for coindexation on PT. However, other parsers have much worse results, with [17] being the worst, with 68 % F-score. The presence of additional difficulties to predict empty categories is the cause of a bad drop in performance in Chinese—no more than 50 % accuracy reported by [4] compared to 74/77 % of the labeled/unlabed task. Results reported by Yang & Xue [27] on recovering labeled empty elements in an experiment carried on a small subset of the Penn Chinese Treebank 6.0 [26] reach an average of 60.5 % of F-measure. As to

recovery of specific items, we note that over a total number of 290 little_pro items recall fares around 50 %.

In addition to that problem, PT annotators made it clear from the start that coindexation is not performed with adjuncts structure which are difficult to judge. In fact, in Sect. 4.8.8 of the Bracketing Guideline [1], under the header Limits of coindexation, the authors comment on the problem of annotators' agreement and formulate rules for not coindexation as follows,

"The * null element generally receives a reference index whenever there is an appropriate referent elsewhere in the same sentence. However, there are cases in which annotators tend to not coindex, even when they can find a plausible referent…. (1) Annotators usually avoid indexing from non-arguments…. (2) Null subjects of gerund complements of PP modifiers of NPs are coindexed only if there is a particularly strong coindexed interpretation or the PP appears to be part of some "fixed phrase"…. (3) NP brackets that only mark a possessive phrase within an NP should NOT serve as a referent for a * null element."

Rule (1) is exemplified by example (1) as follows,

(1) For Willie, it is difficult to resist chocolate.

```
(S (PP For
   (NP Willie))
  ,
  (NP-SBJ (NP it)
     (S *EXP*-1))
  (VP is
    (ADJP-PRD difficult)
    (S-1 (NP-SBJ *)
      (VP to
        (VP resist
          (NP chocolate))))))
```

where we see a fronting of the FOR PPphrase. This should be treated differently from the argument case shown in the example below,

(2) It was impossible for anyone to escape.

```
(S (NP-SBJ (NP It)
     (SBAR *EXP*-1))
     (VP was
           (ADJP-PRD impossible)
           (SBAR-1 for
           (S (NP-SBJ anyone)
                 (VP to
                   (VP escape))))))
```

Producing this structure would be hard for statistical parsers [24], since they should be able to distinguish infinitivals argument from non argument ones: in fact, both structures are treated in the same way by Stanford's parser [28].

Also rule (3) is clearly too restrictive for statistical parsers: these are cases of SUBJect controlled infinitivital complements headed by a deverbal noun with an internal possessive or genitive. Whereas in the corresponding sentence containing the deverbal noun acting as a verb and the genitive or possessive as a subject, this would be correctly coindexed with the subject of the infinitival, in the nominalized version of PT they are not as shown in,

```
(3) John's decision to leave
(NP (NP John 's)
     decision
     (S (NP-SBJ *)
        (VP to
         (VP leave))))
(3a) I made a decision to leave
   (S (NP-SBJ I)
     (VP made
      (NP a decision
      (S (NP-SBJ *)
        (VP to
         (VP leave))))))
```

Here the noun DECISION derives from DECIDE and as such it requires control of the SUBJect of the infinitival by the matrix SUBJect. These cases should have been treated in the same manner as in examples below where the SUBJect is introduced by FOR

```
(4) a movie for us to see
(NP (NP a movie)
   (SBAR (WHNP-3 0)
      for
      (S (NP-SBJ us)
        (VP to
          (VP see
            (NP *T*-3))))))

(4a) a good way for them to do it
   (NP (NP a good way)
    (SBAR (WHADVP-4 0)
      for
      (S (NP-SBJ them)
        (VP to
          (VP do
            (NP it)
            (ADVP-MNR *T*-4))))))
```

Anyway, this is still a valuable piece of information for the final representation to be retained for semantic processing. But as said above, this is also erased in

order to allow for smooth machine learning to work properly. As a result, current statistical or probabilistic parsers don't include any null element. With the exception of Stanford's parser which produces a few cases of SUBJect null elements—in the conversion from constituent to typed dependency—that include some argumental infinitivals and PASSIVE—the latter being a controversial issue, considering that chomskian theory treats it as leaving a trace, and LFG just looks for morphological and lexical features.

Predicate-Argument Structures (PASs) can be related to Ternary Expressions introduced in the field of the Semantic Web (hence SW) and used by such researcher like [19] and [21]. They have started to work in the direction of using NLP to populate a database of RDFs, thus creating the premises for the automatic creation of ontologies to be used in the SW. People have come to believe that the problem of NLP might be reduced to that of creating ternary expressions; in turn the problem of ontologies has also been reduced to that of having ternary expressions available. This reduction is in our opinion absolutely misleading and not to further: we want to make it clear that in no way RDFs and ternary expressions may constitute a formal tool sufficient to express the complexity of natural language texts.

RDFs are assertions about the things (people, Webpages and whatever) they predicate about by asserting that they have certain properties with certain values. If we may agree with the fact that this is natural way of dealing with data handled by computers most frequently, it is also a fact that this is not equivalent as being equally useful for natural language. The misconception seems to be deeply embedded in the nature of RDFs as a whole: they are directly comparable to attribute-value pairs and DAGs which are also the formalism used by most recent linguistic unification-based grammars. From the logical and semantic point of view RDFs also resemble very closely first order predicate logic constructs: but we must remember that FOPL is as such insufficient to describe natural language texts. Ternary expressions(T-expressions), <subject relation object>.

Certain other parameters (adjectives, possessive nouns, prepositional phrases, etc.) are used to create additional T-expressions in which prepositions and several special words may serve as relations. For instance, the following simple sentence

(5a) Bill surprised Hillary with his answer

will produce two T-expressions:

(5b) <<Bill surprise Hillary> with answer> <answer related-to Bill>

In Litkowski's system the key step in their question-answering prototype was the analysis of the parse trees to extract semantic relation triples and populate the databases used to answer the question. A semantic relation triple consists of a discourse entity, a semantic relation which characterizes the entity's role in the sentence, and a governing word to which the entity stands in the semantic relation. The semantic relations in which entities participate are intended to capture the semantic roles of the entities, as generally understood in linguistics. This includes such roles as agent, theme, location, manner, modifier, purpose, and time. Surrogate place holders included are "SUBJ", "OBJ", "TIME", "NUM", "ADJMOD", and the prepositions

heading prepositional phrases. The governing word was generally the word in the sentence that the discourse entity stood in relation to. For "SUBJ", "OBJ", and "TIME", this was generally the main verb of the sentence. For prepositions, the governing word was generally the noun or verb that the prepositional phrase modified. For the adjectives and numbers, the governing word was generally the noun that was modified.

People working advocating the supremacy of the TEs approach were reacting against the Bag of Words approach of IR/IE in which words were wrongly regarded to be entertaining a meaningful relation simply on the basis of topological criteria: normally the distance criteria or the more or less proximity between the words to be related. Intervening words might have already been discarded from the input text on the basis of stopword filtering. Stopword lists include all grammatically close type words of the language considered useless for the main purpose of IR/IE practitioners seen that they cannot be used to denote concepts. Stopwords constitute what is usually regarded the noisy part of the channel in information theory. However, it is just because the redundancy of the information channel is guaranteed by the presence of grammatical words that the message gets appropriately computed by the subject of the communication process, i.e. human beings. More on this topic below.

1.1 Machine Translation and Q/A will Benefit from Null Elements

Computing complete Predicate-Argument structures (PASs) is essential for Machine Translation tasks—as [6] have shown—where one of the two languages belongs to typology above. As an example, we tried out the translation of one sentence from Italian into English, where I introduced null elements and lexical pronouns. We used Personal Translator, Systran and Google online translation websites. I marked null elements with italics: there 3 null subjects of tensed clause (little_pro), one null subject of untensed clause (PRO), one enclitic pronoun (la), two possessive pronouns (sua, propria):

(6) Maria successivamente, dopo aver rifiutato la sua offerta, gli ha detto che vuole vendere la propria casa a sua sorella perché vuole aiutarla.

Maria successivamente, dopo *PRO* aver rifiutato la *sua* offerta, *little_pro* gli ha detto che *little_pro* vuole vendere la *propria* casa a *sua* sorella perché *little_pro* vuoleaiutarla.

Gold Translation
Then, after having rejected his offer, Maria told him that she intends to sell her (own) house to her sister because she wants to help her.
Google Translation
Maria later, after she refused his offer, told him he wants to sell his house to his sister because she wants to help.

Systran Translation
Maria successively, after to have refused its offer, she has said it that she wants to sell own house to its sister because she wants to help.
Personal Translator
Maria subsequently, after refusing his offer, told he wants to sell its house his sister because he wants to help her.

The sentence is farly simple both in lexical choice and syntactic structure. As can be gathered, Google makes grammatical mistakes due to lack of long distance control—"he, his, his" are all in masculine gender rather than feminine. Systran gets the subject empty pronouns right, but then mistakes the possessives—"its" is neutral—and uses unfrequent adverbials like "successively" to translate "dopo". Of course, Google gets an overall best translation both on grammatical and lexical side. None of the translation includes the object enclitic "-la"/her in the output: in fact, the verb "help" can be used intransitively, i.e. omitting the object and no mistake ensues. However in this way the left over pronoun is implicit and needs to be evoked. If we substitute "aiutarla" with "lasciarla" we obtain two different behaviours. Google produces the same output: no pronoun. In this case, however the meaning is no longer preserved and "she wants to leave" has a totally different meaning from "she wants to leave her". Systran on the contrary produces "it" for singular no matter what gender it is ("lo", "la"), and "them" for plural.

Finally, Personal Translator has a fairly understandable translation where however pronouns are mostly wrong—its, he—; however, this is the only system that manages to translate the enclitic pronoun la/her at the end of the sentence, and does that correctly.

Now consider simple questions like "What do mice eat?" versus "What eats mice?" This is called the Reversible Argument Problem [11]. The verb "eat" entertains asymmetrical relations with its SUBJect and its OBJect: in one case we talk of the "eater", the SUBJect and in another case of the "eatee", the OBJect. The asymmetry of relation in transitive constructions involving verbs of accomplishments and achievements (or simply world-changing events) is however further complicated by a number of structural problems which are typically found in most languages of the world, the first one and most common being Passive constructions:

(7i) John killed Tom.
(7ii) Tom was killed by a man.
And the question "Who killed the man?"

Answer to the question would be answered by "John" in case the information available was represented by sentence in i., but it would be answered by "Tom" in case the information available was represented by sentence ii. Obviously this would happen only in lack of sufficient NLP elaboration: a too shallow approach would not be able to capture presence of a passive structure. BOWs approaches only consider proximity between two keywords or entities but not their reciprocal order. There is a certain number of other similar structure in texts which must be regarded as inducing into the same type of miscomputation: i.e. taking the surface order of NPs as indicating the deep intended meaning. In all of the following constructions the

surface subject is on the contrary the deep object thus the Affected Theme or argument that suffers the effects of the action expressed by the governing verb rather than the Agent:"*Inchoatized structures; Ergativized structures; Impersonal structures*. Other important and typical structures which constitute problematic cases for a surface chunks based approach to text computation are the following ones in which one of the arguments is missing and Control should be applied by a governing NP, they are called in one definition Open Predicative structures and they are: "*Relative clauses; Fronted Adjectival adjunct clauses; Infinitive clauses; Fronted Participial clauses,; Gerundive Clauses; Elliptical Clauses; Coordinate constructions*". These structures are discussed at length in the sections below and need proper coindexation between the controller, a Subject or Object noun and the Null Element. It is just coindexation that will allow the system to substitute the pronoun with its antecedent and thus produce a complete PAS.

2 The Rule-Based Algorithm for Empty Elements

We present a symbolic rule-based algorithm that takes as input CONLL-style dependency based representations and populates them with all lexically unexpressed and implicit linguistic elements. We have been working with two languages, Italian and English, but we assume that the algorithm can be applied to any language provided a subcategorization computational lexicon is available for the language. The algorithm also computes best semantic roles to associate to arguments and adjuncts, and provides antecedents for all types of controlled empty subjects. It also makes use of a pronominal anaphora resolution algorithm which however only gives a preference antecedent that requires manual checking. But then if we read through the Bracketing Guidelines [1] we discover that for the majority of the cases null elements have been annotated without indicating the coindexed lexical item, and this is in particular true for all the adjuncts structures which need them, i.e. gerundives and infinitivals.

We tested the algorithm on a fragment of VIT, the Venice Italian Treebank, which contains 500 sentences and 15,000 tokens and we ended up with an addition of over 600 new items fully coindexed. Providing unexpressed and implicit linguistic items is a paramount process to enable semantic predicate argument representations to be produced automatically [5]. This is not only an essential step for a complete linguistic resource such as a treebank, but also for any follow up, be it MT or Question Answering where there is a need for fully implemented predicate-argument structures.

We present an algorithm that starting from a surface dependency parsing of a text in CONLL style populates the representation with the missing implicit or unexpressed linguistic elements:

- these can be unexpressed SUBJects of untensed clauses (including infinitivals, participials, gerundives be they computable as arguments or as adjuncts of a given predicate);

- unexpressed SUBJects of tensed clauses, this being highly language specific—whereas Italian freely allows to leave unexpressed the subject of tensed clause, English will only allow it in imperatives and coordinate clauses;
- traces, or empty linguistic items in what are called "long distance dependency" constructions, like relative clauses and interrogative clauses;
- for every new added empty linguistic item, the algorithm will look for the antecedent to which the item will be dependent—this can be local for most of the cases, but it can also be external to the sentence where the empty item has been added. In this latter case, then the antecedent can be definite and fully referential, or it can be indefinite or better generic, thus non referential. This applies to impersonal pronouns, to untensed clauses with generic reference.
- in the case of Italian, SUBJects of tensed classed will search for the antecedent in a previous stretch of discourse with an anaphoric binding algorithm that builds a history list and computes best antecedents on the basis of semantic features associated to each referring expression computed in the current sentence.

2.1 PTB Conversion is Hardly Ever Ideal

We have been referring to CONLL style column representation used in the CONLL shared tasks series of conferences which are a conversion of Penn Treebank portions by means of Lund's University tool. In fact, the conversion contains many mistakes which badly ruin the semantic import of the output. In this section we shall comment on some examples before presenting our algorithm.

All examples are taken from portion 24 of Penn Treebank and have been produced with Lund's converter. One of the problems of the conversion, and indirectly of all CONLL shared tasks that use these representations, is the way in which Relative∼Clauses are built. In order to do away with the need to add empty traces, the relative pronoun or complementizer is made dependent directly to the verb of the relative clause: here, the pronoun WHO is directly interpreted as the SUBJect of the verb BE that follows it and the head noun LIONS is the head of the verb of the relative clause. In this way the relative pronoun is not part of a chain that goes from the head noun to the empty coindexed argument or adjunct in the following relative clause, as indicated in PT, that we attach below:

(5) the so-called young lions who are anxious to see the old lions in action ...
23 the _ DT _ _ 26 NMOD _ _
24 so-called _ JJ _ _ 26 NMOD _ _
25 young _ JJ _ _ 26 NMOD _ _
26 lions _ NNS _ _ 21 APPO _ _
27 who _ WP _ _ 28 SBJ _ _
28 are _ VBP _ _ 26 NMOD _ _
29 anxious _ JJ _ 28 PRD _ _
30 to _ TO _ 29 AMOD _ _

```
31 see _ VB _ _ 30 IM _ _
32 the _ DT _ _ 34 NMOD _ _
33 old _ JJ _ _ 34 NMOD _ _
34 lions _ NNS _ _ 31 OBJ _ _
35 in _ IN _ _ 31 LOC-OPRD _ _
36 action _ NN _ _ 35 PMOD _ _

(NP
 (NP (DT the) (JJ so-called) (JJ young) (NNS lions) )
  (SBAR
    (WHNP-2 (WP who) )
      (S
       (NP-SBJ (-NONE- *T*-2) )
        (VP (VBP are)
         (ADJP-PRD (JJ anxious)
          (S
           (NP-SBJ (-NONE- *) )
            (VP (TO to)
             (VP (VB see)
              (S
               (NP-SBJ (DT the) (JJ old)
                           (NNS lions) )
                (PP-LOC-PRD (IN in)
                  (NP (NN action) ))))))))))))))))))))
```

This is clearly the opposite of what is usually the case in main clause, where the complements and the subject are dependent on the verb and not the other way around. Predicate-Argument structure of the verb of the relative requires a SUBJect to be expressed and substituted to the head of the pronoun WHO: there is no explicit information available that WHO should be substituted by LION its head noun. Two important difficult and additional cases are constituted by those relative pronouns which do not make arguments in the relative clause but only adjuncts: here is a genitive “WHOSE” and a locative “IN WHICH”:

(6) Now Sony, whose innovative, premium-priced products are among the most admired...

```
1 Now _ RB _ _ 18 TMP _ _
2 Sony _ NNP _ _ 18 SBJ _ _
3 , _ , _ _ 2 P _ _
4 whose _ WP$ _ _ 8 NMOD _ _
5 innovative _ JJ _ _ 8 NMOD _ _
6 , _ , _ 8 P _ _
7 premium-priced _ JJ _ _ 8 NMOD _ _
8 products _ NNS _ 9 SBJ _
9 are _ VBP _ _ 2 NMOD _ _
10 among _ IN _ _ 9 LOC-PRD _ _
```

11 the _ DT _ _ 13 NMOD _ _
12 most _ RBS _ _ 13 AMOD _ _
13 admired _ VBN _ _ 10 PMOD _ _

```
( (S
  (ADVP-TMP (RB Now) )
  (NP-SBJ
   (NP (NNP Sony) )
   (, ,)
   (SBAR
    (WHNP-1 (WP$ whose) (JJ innovative)
     (, ,)
     (ADJP (JJ premium-priced) )
     (NNS products) )
    (S
     (NP-SBJ (-NONE- *T*-1) )
     (VP (VBP are)
      (PP-LOC-PRD (IN among)
       (NP
        (NP (DT the)
         (ADJP (RBS most) (VBN admired) ))
        (PP-LOC (IN in)
         (NP (NN consumer) (NNS electronics) ))))))))
```

The subject of "are among" has wrongly become SONY. However the relevant point is that the relative pronoun is missing its internal trace as a genitive to the head noun PRODUCTS (whose=of Sony). The same happens in the following sentence, where the locative relative pronoun IN WHICH is linked to the relative clause verb and also its head noun SCENE, but since there is no dependency link between the relative pronoun and the head noun it will be hard to determine the function, let alone the role.

(7) a marvelously cute scene in which the trading-room crew minded a baby, the casualty of a broken marriage at the firm.
6 a _ DT _ _ 9 NMOD _ _
7 marvelously _ RB _ _ 8 AMOD _ _
8 cute _ JJ _ _ 9 NMOD _ _
9 scene _ NN _ _ 5 PMOD _ _
10 in _ IN _ _ 15 LOC _ _
11 which _ WDT _ _ 10 PMOD _ _
12 the _ DT _ _ 14 NMOD _ _
13 trading-room _ NN _ _ 14 NMOD _ _
14 crew _ NN _ _ 15 SBJ _ _
15 minded _ VBD _ _ 9 NMOD _ _
16 a _ DT _ _ 17 NMOD _ _
17 baby _ NN _ _ 15 OBJ _ _

18 , _ , _ _ 17 P _ _
19 the _ DT _ _ 20 NMOD _ _
20 casualty _ NN _ _ 17 APPO _ _
21 of _ IN _ _ 20 NMOD _ _
22 a _ DT _ _ 24 NMOD _ _
23 broken _ VBN _ _ 24 NMOD _ _
24 marriage _ NN _ _ 21 PMOD _ _
25 at _ IN _ _ 24 LOC _ _
26 the _ DT _ _ 27 NMOD _ _
27 firm _ NN _ _ 25 PMOD _ _
28 . _ . _ _ 4 P

(NP
(NP (DT a)
 (ADJP (RB marvelously) (JJ cute))
  (NN scene))
(SBAR
 (WHPP-2 (IN in)
 (WHNP (WDT which)))
  (S
 (NP-SBJ (DT the) (NN trading-room) (NN crew))
   (VP (VBD minded)
    (NP
     (NP (DT a) (NN baby))
      (, ,)
     (NP
      (NP (DT the) (NN casualty))
       (PP (IN of)
        (NP
       (NP (DT a) (VBN broken) (NN marriage))
          (PP-LOC (IN at)
           (NP (DT the) (NN firm)))))))
      (PP-LOC (-NONE- *T*-2))))))))

Other important attempts at using PT constituent representation to convert it into a semantically consistent structure has been carried out by research of two other linguistic theories, CCG [8, 9] and LFG [16]. In particular, in [2] an experiment is reported by which the authors produced a fully converted version of section 23 of PT into f-structure representation. At a quick perusal of the output which can be downloaded from their website, however, one notices that the null elements with no coindexation have been introduced in the f-structure without providing an antecedent. This blurs the resulting semantics, seen that f-structures don't work with empty variables, since they are very much like logical forms [3], as shown in example (8) their gold_77,

(8):He said he believes GM has plans to keep building A-body cars into the mid-1990s .
subj : pred : He
comp : subj : pred : he
comp : subj : num : sing
pers : 3
pred : GM
obj : relmod : xcomp : subj : _7499
xcomp : subj : _7499
obj : adjunct : 1 : pred : A-body
pred : cars
num : pl
pers : 3
participle : pres
pred : building
adjunct : 2 : obj : spec : det : pred : the
pred : into
pred : keep
subj : _7499
to : +
inf : +
num : pl
pers : 3
pred : plans
tense : pres
pers : 3
num : sing
pred : has
tense : pres
pers : 3
num : sing
pred : believes
tense : past
pred : said

The most important attempt at using PT constituent representation to convert it into a semantically consistent structure has been carried out by PARC 700 Xerox group. The corpus consists of the usual section 24 of PT and is freely downloadable. Here we look at some examples illustrating the way in which WHOSE is annotated. We only report the relevant portion of the LFG f-structure representation, where it is clearly apparent that the treatment is definitely organized on the basis of the presence of a NULL element, an abstract *"pro"*. What is important to stress here is the fact that WHOSE expresses a possessive genitive relation with its local head that it modifies, and that this relation is represented by *"pro"* linked to WHOSE which in turn is in

a chain with the head noun, and then linked to the verb of the relative, in the three excerpts examples, BE, DETERMINE, KEEP:

id(wsj_2369.35, parc_23.548)

sentence_form(And it has remained there\ , as evidenced by its reappearance in a 1972 CBS sitcom called ‘‘Bridget Loves Bernie\ ,’’ whose sole distinction was that it led to the real-life marriage of Meredith Baxter and David Birney.)

subj(call∼18, pro∼26)
subj(Bridget Loves Bernie∼25, pro∼26)
xcomp(call∼18, Bridget Loves Bernie∼25)
adjunct_type(be∼19, relative)
subj(be∼19, distinction∼31)
topic_rel(be∼19, distinction∼31)
pron_rel(be∼19, pro∼32)
pron_form(pro∼32, whose)
pron_type(pro∼32, relative)
poss(distinction∼31, pro∼32)

id(wsj_2384.44, parc_23.596) sentence_form(The White House Office of Management and Budget\ , whose calculations determine whether the Gramm-Rudman targets are met\ , estimated that the House-passed deficit-reduction measure would cut the fiscal 1990 shortfall by $6.2 billion\ , almost half of the Congressional Budget Office’s estimate of $11.0 billion.) adjunct(Office of Management and Budget∼4, determine∼26)
adjunct_type(determine∼26, relative)
pron_rel(determine∼26, pro∼33)
subj(determine∼26, calculation∼31)
poss(calculation∼31, pro∼33)
pron_form(pro∼33, whose)
pron_type(pro∼33, relative)

id(wsj_2343.17, parc_23.685)

sentence_form(Her friend Susan\ , whose parents kept reminding her she was unwanted\ , slept on a narrow bed wedged into her parents’ bedroom\, as though she were a temporary
adjunct(Susan∼1, keep∼45)
mod(Susan∼1, friend∼61)
adjunct_type(keep∼45, relative)
pron_rel(keep∼45, pro∼48)
subj(keep∼45, parent∼49)
topic_rel(keep∼45, parent∼49)
pron_form(pro∼48, whose)

pron_type(pro∼48, relative)
poss(parent∼49, pro∼48)

3 VIT Description

The VIT Corpus consists of 60,000 words of transcribed spoken text and of 270,000 words of written text. In this chapter I will restrict my description to the characteristics of written texts of our Treebank. We presented lately [10, 25] an algorithm for the automatic conversion of VIT, which uses traditionally bracketed syntactic constituency structures, into a linear word- and column-based head-dependent representation enriched with grammatical relations, morphological features and lemmata.

We organized our work into a pipeline of intermediate steps that incrementally carried out the full conversion task. In this way we also managed to check for consistency at different levels of representation.

The fully converted file also includes Grammatical Relation labels and some Semantic Role, related to Locative and Manner complements and adjuncts. Content words have also been enriched with semantic class information and morphological features coming from our morphological analyser which provided also lemmata. In a language like English, which imposes a strict position for SUBJect NP and OBJect NP, the labeling is quite straightforward. The same applies also to French, and German, which in addition has case markings to supplement constituent scrambling, i.e. the possibility to scramble OBJect and Indirect OBJect in a specific syntactic area.

As opposed to these and other similar languages, which are prevalent in Western language typology, Italian is an almost "free word-order" language—deriving from Latin and strongly influenced by it. In Italian, non-canonical positions would indicate the presence of marked constructions—which might be intonationally marked—containing linguistic information that is "new", "emphasized" or otherwise non-thematic. Italian also allows free omission of a SUBJect NP whenever it stands for a discourse topic. Italian also has lexically empty non-semantic expletive SUBJects for impersonal constructions, weather verbs etc.

We wanted to highlight difference between canonical and non-canonical arguments, seen that it might well turn out that number of non-canonical arguments constituted a high percentage. We thus started to relabel non-canonical SUBJect and OBJect NPs, with the goal of eventually relabeling all non-canonical arguments. However, we realized that we could maintain a distinction between SUBJects on the one side and complements in general on the other, where the former can be regarded as *external* arguments, receiving no specific information at syntactic level from the governing predicate to which they are related. Arguments that are complements are, in contrast, strictly *internal* and are directly governed by predicates, whether the latter are Verbs, Adjectives or Nouns. Eventually, non-canonical Subjects were given three different labels according to their position, whereas other complements were only marked LDC in case they preceded rather than followed their governing predicate.

Prepositions constitute a special case in that they govern PPs which are exocentric constituents and are easily relatable to the NP head they govern. However, it must be possible to relate PPs to their governing predicate, which may or may not subcategorize for them, according to Preposition type. A similar question is related to the more general need to tell apart arguments and adjuncts in ditransitive and intransitive constructions. In Italian, prepositional phrases can occur quite freely before or after another argument/adjunct of the same predicate. So it is impossible to automatically mark ditransitive PP complements without subcategorization information, or mark PPs as OBLiques without appropriate semantic and lexical information.

The solution to this problem was on the one hand the use of our general semantically labeled Italian lexicon which contains 17,000 verb entries together with a lexicon lookup algorithm, where each verb has been tagged with a specific subcategorization label and a further entry for prepositions for which it subcategorizes. The use of this lexicon has allowed the automatic labelling of PP arguments in canonical positions and reduced the task of distinguishing arguments from adjuncts to the manual labeling of arguments in non-canonical positions.

On the other hand, as nominal heads were tagged with semantic labels, we proceeded to label possible adjuncts related to space and time. With verbs of movement, where the subcategorization frames required and the preposition heading the PP allowed it, we marked the PP as argument. We also relabeled as arguments all those PPs that were listed in the subcategorization frames of Ditransitives, again where the preposition allowed it.

The process included the following steps. First, we manually listed all S_DIS (preposed subject under CP), S_FOC (focalized object/subject in inverted position, no clitic), S_TOP (topicalized subject/object to the right, with clitic) and LDC (left dislocated complement, usually SA/SQ/SN/SP/SPD/SPDA) structures.

The resulting treebank has now 10,607 constituents with a subject role, 3,423 of which have been assigned manually because they are in a non-canonical position. Among the 7,184 SUBJ labels that were automatically identified, 46 constituents should have been assigned a different function, which means that we reached the precision of 0.99. On the other hand, 218 constituents should bear a SUBJ label instead of their actual label, which means that the value for recall is 0.97.

If one considers that in PT there are 93,532 sentence structures—identifiable using the regular expression "(S (" – 38,600, or 41% of which are complex sentences, the cases of non-canonical SUBJect occur in only about 1 % of the cases. By contrast, in VIT the same phenomenon has a much higher incidence: over 27 % for non-canonical structures, and over 50 % for the omitted or unexpressed subject. Table 1 also takes into consideration the annotation of complements in non-canonical positions.

Table 2 shows absolute values for all non-canonical structures we relabeled in VIT. There were 7,172 canonical lexically expressed SUBJects out of the 10,100 total expressed SUBJects, which means that non-canonical subjects constituted 1/3 of all expressed SUBJects. Subject NPs positioned to the right of the governing verb were labeled S_TOP. Subject NPs positioned to the left of the governing verb but separated from it by a heavy or parenthetical complement were labeled as S_DIS. S_FOC was

the label used for subjects in inverted postverbal positions in presentational structures. Finally LDC is the label for left dislocated complements with or without a doubling clitic.

4 Creation of Null Elements

Eventually VIT looked very similar to the output of current state-of-the-art statistical treebank parsers trained on PTB [20]. So we imagined that we could create a script or algorithm to try and produce all null elements and try to coindex them automatically, in line with what other researchers have done for Chinese, for example which has similar problems—left-dislocation and unexpressed subject, in particular, in addition Italian has also right dislocation and clitics [12, 13]. We selected 500 complex sentences from VIT, with average sentence length of 30 tokens, total tokens 15,000. However, before starting work on the algorithm, we realized soon that some ambiguity had to be solved manually or else our automatic procedure would never be able to come to a reasonable solution. I am referring to a manual classification of SI (pro)clitic—as in a sentence like "qui si mangia bene"/Here you can eat well where SI appears with a generic impersonal meaning—which is a cause of difficulty even for the most skilled annotators. When we worked at the construction of the annotation manual for ISST national project for the Italian treebank, we came up together with colleagues from Pisa unit to the following fine-grained classification for SI:

- "si" passivizing, diat=middle, reflex=passive
- impersonal "si", diat=active
- intransitive pronominal, with "si", diat=middle, reflex=ipron
- reflexive, with "si", diat=middle, reflex=rifl
- reflexive apparent, diat=middle, reflex=rifl_app
- reflexive apparent as in "ci_si", diat=middle, reflex=rifl_app
- reflexive as in "ci_si", diat=middle, reflex = rifl

We then eventually agreed on what is computationally relevant, that is the distinction between "impersonal_si". "reflexive SI", and "expletive or pleonastic SI". These

Table 1 Comparison of non-canonical Structures in VIT and in PTB where we differentiate TU (total utterances) and TS (total simple sentences)

PT and VIT versus NC Strucs.	NC Strucs. (TU)	Structs. with NC Subject (TS)	Total (TU) Utteran.	Total (TS) Simple Sents	Total Compl. Sents
VIT	3,719	9,800	10,200	19,099	6,782
Percent	27.43%	51.31%	63.75%		66.5%
PT	7,234	2,587	55,600	93,532	38,600
Percent	13.01%	0.27%	59.44%		69.4%

Table 2 Non-canonical Structures in VIT : LDC=left dislocated complements, S_DIS=dislocated subject, S_TOP=topicalized subject, S_FOC=focalized subject

Type of. Struc.	Freq. occurr.
LDC	251
S_DIS	1,037
S_TOP	2,165
S_FOC	266
Total Non-can	3,719

Table 3 Little_pros in portion of VIT

Type of. Real.	Freq. occurr.
Discourse	70
subj_expl	47
subj_impers	38
subj_impl+ant	65
Total little_pro	223

three cases have however to be distinguished manually. Differentiating "middle" cases would be beneficial for Semantic Role assignment because it is always the case that the deep object has been raised to become the subject. However, introducing this additional feature would have made the classification impossible to complete in a short period of time.

After completing this work we went back to the algorithm which is organized in different steps as follows: the first step has been the annotation of all missing subject of tensed clauses, what is usually called the little_pro instance of empty subject pronoun. This is clearly a preliminary step in that it is then mandatory to complete the argument structure of each clause before dealing with "untensed" clauses, that is infinitivals, participials and gerundives. This process is itself organized as the addition of a null element with the same index of the governing verb, which was then diversified by the association of an additional number, 11. Then we wanted to add features coming from the antecedent and from the verb; the real problem then was finding the antecedent: to that aim we recovered our anapahora resolution algorithm and adapted it to the task. But then we discovered that only a percentage of all little_pros required an anaphora resolution algorithm, 31.4 %. The remaining cases had local antecedents of different types or were simply expletive subjects, as shown in Table 3. below.

The examples below illustrate the output of the manual and automatic annotation: we introduced for verbs both a fine-grained syntactic category and a semantic class taken from our subcategorized lexicon; for arguments and adjuncts we added semantic roles by a bottom up procedure that chose the best frame according to available information. Here are some excerpts of the new updated VIT related though to different null elements classified:

Case 1. Impersonal Subject
…quando si arriva/when one arrives
18 quando quando cosu fs [] 20 fs temp
19 si si clit ibar per=3|gen=m|num=sp 20 ibar nom
20 arriva arrivare vin ibar punt 30 ibar unac/posit

20.11 pro si little_pro sn per=3|gen=m|num=sp 19 s_impers-theme_unaff nom

Case 2. Implicit Subject with local antecedent
…e dipenderà/and it will depend
11 e e cong fc [] 8 fc sum
12 dipenderà dipendere virin ir_infl punt 11 ir_infl unac/exten
12.11 pro pro little_pro sn num=s|per=3|md='U'|ts='K' ant=1 s_impl-theme_unaff nil

Case 3. Expletive Subject
…ed è in questa quota che/and it is in this share that
12 ed ed cong fc [] 4 fc sum
13 è essere vc ibar punt 12 ibar cop/existence
13.11 pro pro little_pro nil num=s|per=3|md='L'|ts='K' 17 s_expl nil
14 in in preposition sp - 13 pcomp nil
15 questa questo dim sa num=s|gen=f 16 mod nil
16 quota quota noun sn num=s|gen=f 14 pobj com
17 che che complementizer fac - 16 fac nil

Case 4. Expletive Subject with SI antecedent
…si tratta di/it deals with
0 Si si clit ibar - 1 ibar nil
1 tratta trattare vin cl(main) punt - ibar refl/exten
1.11 pro si little_pro nil num=s|gen=m ant=0 s_expl com
2 del di partd spd num=s|gen=m 1 obl det

Case 5. Implicit Subject with relative pronoun antecedent
…Berlusconi che è industriale/Berlusconi who is industrialist
19 Berlusconi Berlusconi nh sn propr 15 s_top-experiencer hum
20 che che rel f2 - 19 binder nil
21 è essere vc ibar punt 23 ibar cop/existence
21.11 pro pro little_pro sn num=s|per=3|md='L'|ts='K' ant=19 s_impl-tema_bound nil
22 industriale industriale noun sn num=s 21 ncomp com

Case 6. Implicit Subject with Discourse antecedent
…annaspa/it fumbles
2 annaspa annaspare vin ibar punt 0 ibar unerg/exten
2.11 pro sisde little_pro sn punt ant=sent_00132/6 s_impl-theme_aff intr

We have six different notations associated with little_pro, which can be bound to impersonal SI, an expletive SI or an extraposed sentential subject, a local antecedent, a relative pronoun as antecedent and finally a discourse level antecedent where the nominal head is reported. In all other cases, morphological features are associated coming either from the verb or from the antecedent itself.

Second step is the recovery of so-called wh- traces in relative and interrogative clauses, otherwise treated as long-distance dependencies in LFG. We found 286 cases of null elements of this type, which we formalize as follows,

Case 7. Implicit Argument/Adjunct with relative pronoun as local antecedent
…concorrenza the si è progressivamente spostata/competition which has increasingly moved
17 concorrenza concorrenza noun sn num=s|gen=f 14 pobj com
18 che che relative f2 - 17 binder nil
19 si si clit ibar per=3|gen=f|num=sp 22 ibar acc
20 è essere ause ibar punt 22 ibar aux
21 progressivamente progressivamente avv ibar [] 22 adjv mn
22 spostata spostare vppin ibar punt 18 ibar refl_in/posit
22.11 rel_pro concorrenza rel_pro bindee num=s|per=3|md='L'|ts='K' ant=17 subj-theme_aff nil

Third step is the recovery of the unexpressed subject of tenseless clauses, which is formalised as big_pPro. We found 139 occurences of this type of null element which is represented with the antecedent index and also the head, as follows:

Case 8. Implicit Subject with local antecedent
…ad aumentare l'efficienze/to increase the efficiency
22 ad ad pt sv2 - 23 sv2 nil
23 aumentare aumentare vit sv2 punt 21 adj tr/exten
23.11 pPro pPro big_pro sn nil ant='10' s_impl-agent infrastruttura
24 l_ il article sn num=s|gen=f 25 sn def
25 efficienza efficienza noun sn num=s|gen=f 23 obj com

Overall we added 617 new fully annotated null elements. Then, we used this dataset as gold data to check the working of the algorithm: we ran the algorithm on the raw version of the dataset and matched the result with the gold augmented version of the dataset of the 500 sentences: we found 43 mistakes (that is 0.7 % error rate), most of which (32, that is 0.5 %) was a wrong antecedent for discourse bound little_pros. Of course this is just a preliminary evaluation which will be extended to the whole of the corpus—comprising 10,200 sentences and 275,000 tokens—in the future.

4.1 Relative Pronouns can be Hard to Compute

In this subsection we will comment on cases of relative pronouns which are very hard to compute. We saw above that the best way to annotate and parse a relative pronoun in dependency structure is in our opinion, the one that treats the relative pronoun—or its substitute THAT/QUE/CHE etc.—as an intermediary element in a chain between the head noun and the verb of the relative clause. Of course, if the structure is enriched with Null Elements, the latter will act as the final slot of the chain, it would receive the relevant grammatical function label, and would be attached to the verb of the relative. However this is not always possible: the examples below show some such hard to compute cases. In Case 1 we have a relative pronoun which is an Adjunct of

an Argument of the Relative Clause; in Case 2, the relative pronoun is in a pied piped or embedded structure, and this will be exemplified also with the output of online parsers.

Case1. Example 1.
"… commissione esteri alla cui presidenza è candidato…"/foreign affairs committee whose presidency is candidate to
8 commissione commissione n(noun) sn num=s|gen=f 6 pobj com
9 esteri estero ag(adjective) sa num=p|gen=m 8 mod nil
10 alla a part(preposition_plus_article) sp num=s|gen=f 8 adj det
10.1 la il art sn num=s|gen=f 8 det def
11 cui cui relob(relative_oblique) sp [] 10 sp rel_obl
12 presidenza presidenza n(noun) sn num=s|gen=f 10 pobj com
13 è essere vc(verb_copulative) ibar punt 8 ibar cop/esistenza
14 candidato candidato n(noun) sn num=s|gen=m 13 ncomp com
14.11 prep_relob alla_commissione prep_relob (prep_rel_oblique) sp num=s| gen=m ant=10_11 bindee com

In this example, we want to say that the relative pronoun modifies CANDIDATO, and the semantics should compose the following pseudo-structure:

commissione esteri [alla cui] presidenza [t] → presidenza [della commissione esteri]

Case1. Example 2.
"Una strategia di cui tutti i ministri interessati continuano a sottolineare la collegialità"/A strategy which all the interested ministers continue to underline the collegiality∼of

0 Una uno art(article) sn num=s|gen=f 1 sn ind
1 strategia strategia n(noun) sn num=s|gen=f 13 sn com
2 di di pd(preposition_di) spd - 1 adj nil
3 cui cui relob(relative_oblique) sn [] 2 binder rel_obl
4 tutti tutto qc(quantifier_collective) sq num=p|gen=m 6 sq nil
5 i il art(article) sn num=p|gen=m 6 sn def
6 ministri ministro n(noun) sn num=p|gen=m 8 subj-exper com
7 interessati interessato ppas(past_participle_absolute) sa num=p|gen=m 6 mod nil
8 continuano continuare vt(verb_trans_tensed) cl(main) punt - ibar raisn/process
9 a a pt(verbal_participle) sv2 - 10 sv2 nil
10 sottolineare sottolineare vit(verb_trans_infinitive) sv2 punt 8 vcomp tr
10.10 pPro pPro pPro(big_pro) sn nil ant='6' s_impl-causer ministro
11 la il art(article) sn num=s|gen=f 13 sn def
12 << par(parenthetical) sn - 13 sn nil
13 collegialità collegialità n(noun) sn num=f 10 obj invar
13.11 prep_relob di_strategia prep_relob (prep_rel_oblique) sp num=s|gen=f ant=1_2 bindee com

In this example, the relative pronoun modifies COLLEGIALITA', and the semantics should compose the following pseudo-structure:

una strategia [di cui] tutti i ministri interessati continuano a sottolineare la collegialità [t] → la collegialità [della strategia]

Case 2.
Not all cases of relative pronouns are connected to a fully lexicalized relative clause: there are cases in which the clause is unexpressed—as would happen with reduced relatives—but also ellipsed as shown in the following examples:

Example 3. "… nomi di rilievo, tra cui l'ex ministro della difesa…"/important names, among which the ex minister of defense
12 nomi nome n(noun) sn num=p|gen=m 11 obj-theme_unaff com
13 di di pd(preposition_di) spd - 12 mod nil
14 rilievo rilievo n(noun) sn num=s|gen=m 13 pobj com
15 , , punt(sentence_internal) sn punt 12 sn nil
16 tra tra p(preposition) sp - 12 adj nil
17 cui cui relob(relative_oblique) sn [] 16 binder rel_obl
18 l il art(article) sn num=s|gen=m 20 sn def
19 ex ex ag(adjective) sa num=f|gen=m 20 mod invar
20 ministro ministro n(noun) sn num=s|gen=m 17 subj com
20.11 prep_relob tra_nome prep_relob (prep_rel_oblique) sp num=p|gen=m ant=16_17 bindee com
21 della di partd(prep_di_plus_article) spd num=s|gen=f 20 mod det
21.1 la il art sn num=s|gen=f 20 det def
22 difesa difesa n(noun) sn num=s|gen=f 21 pobj com

The specialty of this structure is the fact that it is a fragment which however has a main nominal head: to complete the semantics it could be enriched by the presence of a "dummy BE" verb, or perhaps a dummy THERE_BE, so that the head noun MINISTRO is computed as subject of predication. The oblique relative modifies directly the subject nominal MINISTRO or indirectly, in case of presence of dummy BE, through the predication:

→ l'ex ministro … *E' tra i nomi /The ex minister is among the names*

As an experiment I tried out a sentence which contained a pied piped oblique genitive in English, with both CONNEXOR [29] and STANFORD [28] parsers to see the relations they manage to encode in the output. However none of the output is able to show appropriate relations.

Example 4. "John, in whose house the accident took place, is leaving now."
1 John john @OBJ %NH N NOM SG
2 , ,
3 in in @ADVL %EH PREP
4 whose who attr:>5 @A> %>N <Rel> PRON WH GEN
5 house house @<P %NH N NOM SG
6 the the det:>7 @DN> %>N DET
7 accident accident subj:>8 @SUBJ %NH N NOM SG

```
8 took      take       pcomp:>3 @+FMAINV %VA V PAST
9 place     place      obj:>8     @OBJ %NH N NOM SG
10 ,        ,
11 is       be v-ch:>12 @+FAUXV %AUX V PRES SG3
12 leaving leave       @-FMAINV %VA ING
13 home   home     goa:>12 @ADVL %EH N NOM SG
14 now    now      tmp:>12 @ADVL %EH ADV
15 .        .
16 <s>    <s>
```

And this is the STANFORD parser output:

```
(ROOT
 (S
  (NP
   (NP (NNP John))
   (PP (IN in)
    (SBAR
     (WHNP (WP$ whose)
      (NP (NN house)))
     (S
      (NP (DT the) (NN accident))
      (VP (VBD took)
       (NP (NN place)))))))
  (VP (VBZ is)
   (VP (VBG leaving)
    (NP (NN home))
    (ADVP (RB now))))
(. .)))
```

Typed dependencies, collapsed

```
nsubj(leaving-10, John-1)
poss(house-4, whose-3)
dobj(took-7, house-4)
det(accident-6, the-5)
nsubj(took-7, accident-6)
prepc_in(John-1, took-7)
dobj(took-7, place-8)
aux(leaving-10, is-9)
root(ROOT-0, leaving-10)
dobj(leaving-10, home-11)
advmod(leaving-10, now-12)
```

What is missing, then here, is the information that “the house” belongs to John, and the role of WHOSE is left unexplained.

5 Conclusion

I have presented work in progress carried out with the aim of highlighting the importance of Null Elements in Dependency Structures, and their role in the creation of Predicate-Argument structures to be used for the mapping into Logical Form. In turn, Logical Forms are essential representation for any NLP system that intends to use deep semantics for applications like Question/Answering and Information Extraction. Treebanks available today have been discussed and difficulties in producing and annotating them with Null Elements have also been highlighted. I also showed the output of two of the most outstanding online parsers. The chapter focuses then on a proposal to convert currently produced shallow dependency structures into their deep equivalent. This proposal has been preliminarily tested on the Italian treebank VIT and requires computational lexica to contribute deep syntactic and semantic information related to argument structures of predicates, selectional restrictions and other elements made available nowadays in most such linguistic lexical resources, for most major languages. To complete the representation, however, also an algorithm for anaphora resolution has been used. From a preliminary evaluation, results are encouraging but more work needs to be done to cover hard to compute relative clauses and other structures not presented in this chapter. Also results for the anaphora resolution—which are state of the art and average 75 % accuracy—would require further improvements.

Acknowledgments This work has been partially funded by the PARLI Project (Portale per l'Accesso alle Risorse Linguistiche per l'Italiano—MIUR—PRIN 2008).

References

1. Bies, A., Ferguson, M., Katz, K., MacIntyre, R., Tredinnick, V., Kim, G., Ann Marcinkiewicz, M., Schasberger, B.: Bracketing guidelines for Treebank II style Penn treebank. http://www.sfs. uni-tuebingen.de/∼dm/07/autumn/795.10/ptb-annotation-guide/root. html (1995)
2. Cahill, A., McCarthy, M., van Genabith, J., Way, A.: Automatic annotation of the Penn-Treebank with LFG f-structure information. In: LREC: Workshop on Linguistic Knowledge Acquisition and Representation: Bootstrapping Annotated Language Data. Las Palmas (2002)
3. Cahill, A., McCarthy, M., van Genabith, J., Way, A.: Quasi-logical forms for the Penn Treebank. In: Bunt H., van der Sluis I., Morante R. (eds.) Proceedings of the Fifth International Workshop on Computational Semantics, IWCS-05, pp. 55–71. Tilburg (2003)
4. Cai, S., Chiang, D., Goldberg, Y.: Language-independent parsing with empty elements. In: Proceedings of the 49th Annual Meeting of the ACL, pp. 212–216 (2011)
5. Campbell, R.: Using linguistic principles to recover empty categories. In Proceedings of ACL (2004)
6. Chung, T., Gildea, D.: Effects of empty categories on machine translation. In Proceedings EMNLP (2010)
7. Choi, J.D., Palmer, M.: Robust constituent-to-dependency conversion for english. In: Proceedings of the 9th International Workshop on Treebanks and Linguistic Theories (TLT'9), pp. 55–66. Tartu (2010)

8. Clark, S., Curran, J.R.: Comparing the accuracy of CCG and Penn Treebank parsers. In: Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, pp. 53–56. Suntec, Singapore (2009)
9. De Marneffe, M.C., MacCartney, B., Manning, C.D.: Generating typed dependency parses from phrase structure parses. In: Proceedings of LREC, pp. 449–454 (2006/5)
10. Delmonte, R., Bristot, A., Tonelli, S.: VIT —Venice Italian Treebank: Syntactic and Quantitative Features. In: De Smedt, K., Hajic, J., Kübler, S. (eds.), Proceedings of Sixth International Workshop on TLT, vol. 1, pp. 43–54. Nealt Proceeding Series (2007)
11. Delmonte R., Bianchi, D.: Semantic web, RDFs and NLP for QA. In: Calzolari N., Magnini B. (eds.) Proceedings of the Workshop on "Topics and Perspectives of NLP in Italy", Università di Pisa, AI*IA, pp. 67–75 (2003)
12. Dienes P., Dubey, A.: Antecedent recovery: experiments with a trace tagger. In: Proceedingsof EMNLP (2003a)
13. Dienes P., Dubey, A.: Deep processing by combining shallow methods. In: Proceedings of ACL (2003b)
14. Gabbard, R., Marcus M., Kulick, S.: Fully parsing the Penn Treebank. In: Proceedings of the HLT Conference of the North American Chapter of the ACL, pp. 184–191 (2006)
15. Gaizauskas, R.: Investigations into the Grammar Underlying the Penn Treebank II, Technical Report CS-95-25. Univeristy of Sheffield, Department of Computer Science (1995)
16. Guo, Y., van Genabith, J., Wang, H.: Treebank-based acquisition of LFG resources for Chinese. In: Lexical Functional Grammar, pp. 28–30. California (2007)
17. Johnson, M.: A simple patter-matching algorithm for recovering empty nodes and their antecedents. In: Proceedings of the 39th Annual Meeting of the ACL, 136–143, Toulouse, France (2001)
18. Johansson, R., Nugues, P.: Extended constituent-to-dependency conversion for english. In: Proceedings of NODALIDA 2007, Tartu (2007)
19. Katz, B.: Annotating the World Wide Web using natural language. In: RIAO '97 (1997)
20. Liakata, M., Pulman, S.: From Trees to Predicate-Argument Structures. In: Proceedings of the 19th International Conference on Computational Linguistics (COLING 2002), pp. 563–569. Taipei (2002)
21. Litkowski, K.C.: Syntactic clues and Lexical resources in question-answering. In: Voorhees E.M., Harman D.K. (eds.) The Ninth Text Retrieval Conference (TREC-9). NIST Special Publication 500–249, Gaithersburg, pp. 157–166 (2001)
22. Marcus, M., Kim, G., Ann Marcinkiewicz, M., Macintyre, R., Bies, A., Ferguson, M., Katz, K., Schasberger, B.: The Penn Treebank: annotating predicate argument structure. In: ARPA Human Language Technology Workshop, pp. 114–119 (1994)
23. Sagae, K., Tsujii, J.: Shift-reduce dependency DAG parsing. In: Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008), Manchester (2008)
24. Schmid, H.: Trace prediction and recovery with unlexicalized PCFGs and slash features. In: Proceedings COLING-ACL (2006)
25. Tonelli, S., Delmonte, R., Bristot, A.: Enriching the Venice Italian Treebank with dependency and grammatical relations, LREC 2008 (2008)
26. Xue, N., Xia, F., Chiou, F.-D., Palmer, M.: The Penn Chinese TreeBank: phrase structure annotation of a large corpus. Nat. Lang. Eng. **11**(2), 207–238 (2005)
27. Yang, Y., Xue, N.: Chasing the ghost: recovering empty categories in the Chinese Treebank. In: Proceedings COLING (2010)
28. http://nlp.stanford.edu:8080/parser/
29. http://www.connexor.com/nlplib/?q=demo/syntax