Harmonization and Merging of two Italian Dependency Treebanks

Harmonization and Merging of two Italian Dependency Treebanks

Cristina Bosco[*], **Simonetta Montemagni**[◇], **Maria Simi**[†]

[*] Università di Torino, [◇] Istituto di Linguistica Computazionale “Antonio Zampolli” (ILC–CNR) - Pisa, [†] Università di Pisa
bosco@di.unito.it, simonetta.montemagni@ilc.cnr.it, simi@unipi.it

Abstract

The paper describes the methodology which is currently being defined for the construction of a “Merged Italian Dependency Treebank” (MIDT) starting from already existing resources. In particular, it reports the results of a case study carried out on two available dependency treebanks, i.e. TUT and ISST–TANL. The issues raised during the comparison of the annotation schemes underlying the two treebanks are discussed and investigated with a particular emphasis on the definition of a set of linguistic categories to be used as a “bridge” between the specific schemes. As an encoding format, the CoNLL de facto standard is used.

Keywords: Syntactic Annotation, Merging of Resources, Dependency Parsing

1. Introduction

Italian is featured by the availability of four dependency treebanks. Three of them were developed by national research institutions: the Turin University Treebank (TUT)[1] developed by the NLP group of the University of Turin (Bosco et al., 2000); the treebank called ISST–TANL, which was developed as a joint effort by the Istituto di Linguistica Computazionale (ILC–CNR) and the University of Pisa and originating from the Italian Syntactic–Semantic Treebank or ISST (Montemagni et al., 2003); the Venice Italian Treebank (VIT) developed by the University Ca’ Foscari of Venice (Tonelli et al., 2008). A further Italian dependency treebank was developed in the framework of an international project, the Copenhagen Dependency Treebank (Buch-Kromann et al., 2009). Interesting to note, each of these resources, independently developed applying different dependency-based annotation schemes, has a quite limited size, ranging from around 94,000 tokens of TUT to about 60,000 tokens of the Italian CDT section.

In spite of their limited size, some of these resources have successfully been used for training and/or evaluation of dependency parsing systems. For instance, TUT was repeatedly used within the parsing task of the EVALITA evaluation campaign[2] in 2007, 2009 and 2011, for both training and testing dependency parsing systems. A previous version of ISST–TANL, namely ISST–CoNLL, was used for the CoNLL-2007 Shared Task on multilingual dependency parsing as far as Italian is concerned (Nivre et al., 2007; Montemagni and Simi, 2007). ISST–TANL was used in EVALITA 2009 and 2011 for two different tasks, syntactic parsing (Bosco et al., 2009) and domain adaptation (Dell’Orletta et al., 2012) respectively, and is currently being used in the SPLeT 2012 Shared Task on Dependency Parsing of Legal Texts[3].

Despite the encouraging results achieved with these treebanks in the above mentioned initiatives, we are aware that the relatively small size of these resources makes them usable in a restricted variety of tasks with an impact on the reliability of achieved results. By contrast, the availability of a larger resource, harmonizing and merging the original annotated resources, should result in crucial advancements for the Italian NLP.

Preliminary steps in this direction were performed for two of the above mentioned treebanks, namely TUT and ISST–TANL. The first step was represented by the exploitation of these resources in the framework of international evaluation campaigns (CoNLL and EVALITA) which required as a necessary prerequisite the conversion of the native annotation formats into the CoNLL representation standard. A further step was performed in the framework of EVALITA 2009 which included a dependency parsing track (Bosco et al., 2009) articulated into two subtasks differing at the level of used treebanks: TUT was used as the development set in the Main Subtask, and ISST–TANL represented the development set for the Pilot Subtask. The analysis of the results of the best scoring systems, in line with the state of the art dependency parsing technology for Italian, provided the opportunity to start investigating the influence of the design of both treebanks by testing these parsers on a common set of data annotated in both annotation schemes (Bosco et al., 2010). The last and still ongoing step is represented by the national project “Portal for the Access to the Linguistic Resources for Italian” (PARLI), involving several academic NLP groups. PARLI aims at monitoring and coordinating the activities of Italian NLP for fostering the development of new resources and tools that can operate together, and the harmonization of existing ones. The activities carried out within PARLI also comprise the annotation of a new corpus including the full text of the *Costituzione Italiana* [4] by the Pisa and Turin University groups within which the harmonization issue between the TUT and ISST–TANL annotations schemes started to be tackled.

In this paper we describe the methodology we are currently defining for the construction of a “Merged Italian Dependency Treebank” (MIDT) resulting from the harmonization and merging of existing Italian dependency treebanks. This methodology is being tested on the TUT and ISST–TANL treebanks. However, in the near future we would like to ex-

[1] http://www.di.unito.it/˜tutreeb

[2] http://www.evalita.it/

[3] http://poesix1.ilc.cnr.it/splet_shared_task/

[4] http://parli.di.unito.it/activities_en.html

tend this methodology to also cover the other two available Italian dependency treebanks, i.e. VIT and Italian CDT. The paper is organised as follows: after illustrating (Section 2.) the main tenets of our approach to merging, Sections 3. and 4. provide a comparative analysis of the TUT and ISST–TANL annotation schemes, and of the performance of state–of–the–art dependency parsers trained on the two resources. Finally, Section 5. describes the construction of the merged resource and the parsing results achieved by using it as training data.

2. Our approach to merging

Since the early 1990s, different initiatives have been devoted to the definition of standards for the linguistic annotation of corpora with a specific view to re–using and merging existing annotated resources. A first attempt was represented by the outcome of the EAGLES (Expert Advisory Groups on Language Engineering Standards) initiative, in particular of the group of 'experts' set to work on the syntactic annotation of corpora who ended up with providing provisional standard guidelines (Leech et al., 1996). Whereas this first attempt operated at the level of both content (i.e. the linguistic categories) and encoding format, further initiatives tried to tackle these two aspects separately. This is the case, for instance, of LAF/GrAF (Ide and Romary, 2006; Ide and Suderman, 2007) and SynAF (Declerck, 2008), which represent on–going ISO TC37/SC4 standardization activities[5] dealing respectively with a generic meta–model for linguistic annotation and with a meta–model for syntactic annotation, including dependency structures. In both cases, the proposed framework for representing linguistic annotations is intended to be a pivot format capable of representing diverse annotation types of varying complexity which does not provide specifications for annotation content categories (i.e., the labels describing the associated linguistic phenomena), for which standardization appeared since the beginning to be a much trickier matter.

For what concerns the content categories, both architectures include a data category registry containing a (possibly hierarchical) list of data categories meant to represent a point of reference for particular tagsets used for the syntactic annotation of various languages, also in the context of various application scenarios. More recently, this issue is being handled by other standardization efforts such as ISOCat (Kemps-Snijders et al., 2009). ISOCat is intended to provide a set of data categories at various levels of granularity, each accompanied by a precise definition of its linguistic meaning. Labels applied in a user–defined annotation scheme should be mapped to these categories in order to ensure semantic consistency among annotations of the same phenomenon.

The work illustrated in this paper is concerned with the harmonization and merging of dependency–annotated corpora, with a particular emphasis on data categories. As an encoding format, we use the CoNLL representation format, which nowadays represents a *de facto* standard within the parsing community. As far as linguistic categories are concerned, we are not trying to create a single unified annotation scheme to be used by all Italian dependency treebanks: in line with the approaches sketched above, we believe that this represents an impractical and unrealistic task. To put it in other words, it is not a matter about one scheme being right and the other being wrong: we start from the assumption that all schemes are linguistically well–motivated and that there is no objective criterion for deciding which annotation scheme provides the most empirically adequate analysis of the texts. Rather, the challenge we are tackling in this paper, which to our knowledge still represents an open issue in the literature, is to find a way of translating between different annotation schemes and merging them, with the final aim of pooling costly treebank resources. This is being carried out by trying to define a set of linguistic categories to be used as a "bridge" between the specific schemes. This initial effort focused on the TUT and ISST–TANL resources, and in particular on the dependency annotation level, with the long term goal of involving in this process the other available dependency–based Italian treebanks. MIDT, i.e. "Merged Italian Dependency Treebank" represents the final result of the merging process being described in this paper. In order to achieve this goal, we proceeded through the following steps:

- analysis of similarities and differences of considered dependency annotation schemes;
- analysis of the performance of state of the art dependency parsers trained on both treebanks;
- mapping of the individual annotation schemes onto a set of shared (often underspecified) set of data categories;
- last but not least, parametrization of the annotation of the merged resources (still ongoing).

In what follows these different steps are described in detail.

3. The TUT and ISST–TANL treebanks

The TUT and ISST–TANL resources differ under different respects, at the level of both corpus composition and adopted representations.

For what concerns size and composition, TUT currently includes 3,452 Italian sentences (i.e. 102,150 tokens in TUT native, and 93,987 in CoNLL[6]) representative of five different text genres (newspapers, Italian Civil Law Code, JRC-Acquis Corpus[7], Wikipedia and the *Costituzione Italiana*). ISST–TANL includes instead 3,109 sentences (71,285 tokens in CoNLL format), which were extracted from the "balanced" ISST partition (Montemagni et al., 2003) exemplifying general language usage and consisting of articles from newspapers and periodicals, selected to cover a high variety of topics (politics, economy, culture, science, health, sport, leisure, etc.).

As far as the annotation scheme is concerned, TUT applies the major principles of the dependency grammar (Hudson,

[5] http://www.tc37sc4.org/

[6] In the following we will refer only to number of tokens in CoNLL format.

[7] http://langtech.jrc.it/JRC-Acquis.html

1984) using a rich set of grammatical relations, but it includes null elements to deal with non-projective structures, long distance dependencies, equi phenomena, pro drop and elliptical structures[8]. The ISST–TANL annotation scheme originates from FAME (Lenci et al., 2008), an annotation scheme which was developed starting from de facto standards and which was specifically conceived for complying with the basic requirements of parsing evaluation, and – later – for the annotation of unrestricted Italian texts.

3.1. Comparing the annotation schemes

The TUT and ISST–TANL annotation schemes are both dependency-based and therefore fall within the same broader family of annotation schemes. In spite of this fact there are significant differences which make the harmonization and merging of the two resources quite a challenging task. To put it in other words, if on the one hand there is a core of syntactic constructions for which the analysis given by different annotation schemes agree in all important respects, on the other hand there are also important differences concerning the inventory of dependency types and their linguistic interpretation, head selection criteria, the projectivity constraint as well as with respect to the analysis of specific syntactic constructions. In what follows we summarize the main dimensions of variation with a specific view to the merging issues.

Head selection criteria

Criteria for distinguishing the head and the dependent within dependency relations have been widely discussed in the linguistic literature, not only in the dependency grammar tradition, but also within other frameworks where the notion of syntactic head plays an important role. Unfortunately, different criteria have been proposed, some syntactic and some semantic, which do not lead to a single coherent notion of dependency (Kübler et al., 2009). Head selection thus represents an important and unavoidable dimension of variation between the TUT and ISST–TANL schemes, especially for what concerns constructions involving grammatical function words with respect to which there is no general consensus in the tradition of dependency grammar as to what should be regarded as the head and what should be regarded as the dependent. Let us focus on the following tricky cases: namely, the determiner–noun relation within nominal groups, the preposition–noun relation within prepositional phrases, the complementizer–verb relation in subordinate clauses as well as the auxiliary–main verb relation in complex verbal groups.

TUT always assigns heads on the basis of syntactic criteria, i.e. in all constructions involving one function word and one content word (e.g. determiner–noun, preposition–noun, complementizer–verb) the head role is always played by the function word. The only exception is represented by auxiliary–main verb constructions where the head role is played by the main verb. By contrast, in ISST–TANL head selection follows from a combination of syntactic and semantic criteria: i.e. whereas in the determiner–noun and auxiliary–verb constructions the head role is assigned to the semantic head (noun/verb), in preposition–noun and complementizer–verb constructions the head role is played by the element which is subcategorized for by the governing head, i.e. the preposition and the complementizer.

Note that this different strategy in at the level of head selection explains the asymmetric treatment of determiner–noun constructions with respect to preposition–noun ones in ISST–TANL and the fact that for TUT the same dependency type is used for both cases (see below).

Granularity and inventory of dependency types

TUT and ISST–TANL annotation schemes assume different inventories of dependency types characterized by different degrees of granularity in the representation of specific relations. The different degree of granularity of the annotation schemes is testified by the size of the adopted dependency tagsets, including 72 dependency types in the case of TUT and 29 in the case of ISST–TANL. Interestingly however, it is not always the case that the finer grained annotation scheme – i.e. TUT – is the one providing more granular distinctions: whereas this is typically the case, there are also cases in which more granular distinction are adopted in the ISST-TANL annotation scheme. In what follows, we provide examples of both cases.

Consider first TUT relational distinctions which are neutralized at the level of ISST–TANL annotation. A difference in terms of granularity refers e.g. to the annotation of appositive (or unrestrictive) modifiers, which in TUT are annotated by resorting to a specific relation (`APPOSITION`), and which in ISST–TANL are not distinguished from other kinds of modifiers (`mod`). Similarly, TUT partitions predicative complements into two classes, i.e. subject and object predicative complements (`PREDCOMPL+SUBJ` and `PREDCOMPL+OBJ` respectively) depending on whether the complement refers to the subject or the object of the head verb, whereas in ISST–TANL the same dependency type (`pred`) is used to annotate both cases.

Let us consider now the reverse case, i.e. in which ISST–TANL adopts finer–grained distinctions with respect to TUT: for instance, ISST–TANL envisages two different relation types for determiner–noun and preposition–noun constructions (`det` and `prep` respectively), whereas TUT represents both cases in terms of the same relation type (`ARG`). This latter example follows from another important dimension of variation between the two schemes, concerning head selection (see above).

Another interesting and more complex example can be found for what concerns the partitioning of the space of prepositional complements, be they modifiers or subcategorized arguments. TUT distinguishes between `MODIFIER`(s) on the one hand and subcategorised arguments on the other hand; the latter are further distinguished between indirect objects (`INDOBJ`) and all other types of indirect complements (`INDCOMPL`). ISST–TANL neutralizes such a distinction by resorting to a single dependency type, i.e. `comp` (mnemonic for complement), for all relations holding between a head and a prepositional complement, whether a modifier or a subcategorized argument. On the other hand, `comp`(lements) are further

[8] CoNLL format does not include null elements, but the projectivity constraint is maintained at the cost of a loss of information with respect to native TUT (in some cases).

subdivided into semantically oriented categories, such as temporal, locative or indirect complements (comp_temp, comp_loc and comp_ind).

Same dependency type, different annotation criteria
Even when the two schemes show common dependency types, they can diverge at the level of their interpretation, and thus of the underlying annotation criteria. This is the case, for instance, of the "object" relation which in the TUT annotation scheme refers to the direct argument (either in the nominal or clausal form) occurring at least and most once and expressing the subcategorized object, and which in ISST–TANL is meant to denote the relation holding between a verbal head and its non–clausal direct object (other dependency types are used for clausal complements).

Another interesting example is represented by relative clauses. TUT and ISST–TANL follow the same strategy in the representation of standard relative clauses, according to which the head of the relative clause is the verb and the relative pronoun is governed by it as a standard argument. The verbal head is then connected to the antecedent noun through a specific relation, RELCL in TUT and mod_rel in ISST–TANL. However, TUT also treates so–called reduced relative clauses, i.e. constructions where there is no overt relative pronoun and the verb appears in the participial form (either present or past participle), in the same way; namely, by using the same relation type to link the verb of the reduced relative clause to the governing noun. In ISST–TANL, constructions without overt relative pronouns are instead represented by resorting to a general modifier relation (mod).

Projectivity of dependency representations
Projectivity is an important constraint in dependency grammar, relating dependency structures to linear realizations. If on the one hand most NLP systems for dependency parsing assume projectivity, on the other hand this is not the case on the linguistic side where non–projective representations are resorted to for dealing with specific linguistic constructions (e.g. long-distance dependencies) mainly occurring in flexible word order languages (such as Italian). Whereas ISST–TANL corpus allows for non–projective representations, TUT assumes the projectivity constraint.

Treatment of specific constructions
Further important differences between TUT and ISST–TANL annotation schemes are concerned with the treatment of coordination and punctuation, phenomena which are particularly problematic to deal with in the dependency framework.

Besides the general issue widely discussed in the literature of whether coordination can be analyzed in terms of binary asymmetrical relations holding between a head and a dependent, there are different ways put forward to deal with it. In both TUT and ISST–TANL resources, coordinate constructions are considered as asymmetric structures with a main difference: while in ISST–TANL the conjunction and the subsequent conjuncts are all linked to the first conjunct, in TUT the conjuncts starting from the second one are linked to the immediately preceding conjunction.

Also the treatment of punctuation is quite problematic in the framework of a dependency annotation scheme, although this has not been specifically dealt with in the linguistic literature. Both TUT and ISST–TANL schemes cover punctuation with main differences holding at the level of both dependency types and head selection criteria. Whereas ISST–TANL has just one dependency type for all punctuation tokens, TUT distinguishes different dependency types depending on the involved punctuation token and syntactic construction. For example, in TUT an explicit notion of parenthetical is marked while in ISST–TANL it is not. Significant differences also lie at the level of the head assignment criteria: in TUT the head of the punctuation tokens in the parenthetical structure coincides with the governing head of the sub–tree covering the parenthetical structure (i.e. it is external to the parenthetical structure), whereas in ISST–TANL the paired punctuation marks of the parenthetical structure are both connected to the head of the delimited phrase (i.e. internally to the parenthetical).

Other important differences holding between TUT and ISST–TANL schemes are concerned with sentence splitting, tokenization and morpho–syntactic annotation, all aspects which represent important prerequisites for the merging of dependency annotations. All these issues have been addressed and a solution has been proposed as part of the whole harmonization and merging process.[9] In this paper, however, we won't further discuss these aspects and we will focus on the merging of dependency annotations.

4. TUT and ISST–TANL as training corpora

In Bosco et al. (2010), a dependency–based analysis of the performance of state of the art parsers participating in EVALITA 2009 (two stochastic parsers and a rule–based one) with respect to a shared test set was reported, with the final aim of assessing the impact of annotation schemes on parsing results. In particular, for each relation in the TUT and ISST–TANL dependency annotation schemes, the performance of the three parsers was analyzed in terms of Precision (P), Recall (R) and related f-score. In order to identify problematic areas of parsing, both TUT and ISST–TANL dependency–relations were partitioned into three classes (i.e. low-, medium- and best-scored dependency relations) with respect to the associated f-score, which was taken to reflect their parsing difficulty (for more details see Bosco et al. (2010)). Achieved results showed that the improvement of parsing technology should proceed hand in hand with the development of more suitable representations for annotated syntactic data. In this paper we are dealing with the latter issue: we believe that the results of this comparative analysis should also be taken into account in the definition of the merging methodology.

Similar trends were observed in the performance of parsers against TUT and ISST–TANL. First, in both cases hard to parse relations include "semantically loaded" relations such as comp_temp, comp_loc and comp_ind for ISST–

[9] The interested reader is referred to the following URL for more details on the merging of TUT and ISST–TANL morpho–syntactic annotations: http://medialab.di.unipi.it/wiki/POS_and_morphology.

TANL and APPOSITION and INDOBJ for TUT. Moreover, relations involving punctuation appeared to be difficult to parse for statistical parsers in the case of TUT, whereas the rule–based parser had problems dealing with coordinate structures in ISST–TANL; it should be noted however that ISST–TANL con/conj relations show values very close to the low threshold value also in the case of the stochastic parsers. This contrastive analysis thus confirmed a widely acknowledged claim, i.e. that coordination and punctuation phenomena still represent particularly challenging areas for parsing (Cheung and Penn, 2009). The problems raised by the analysis of "semantically loaded" relations in the case of both treebanks suggest that the parsers do not appear to have sufficient evidence to deal reliably with them; in principle, the solutions to the problem range from increasing the size of the training corpus, to neutralising their distinction at this annotation level and postponing their treatment to further processing levels. Concerning the best scored relations, it came out that in both cases they mainly refer to "local" relations. Interesting to note, there is a significant overlapping between the two sets: e.g. the TUT ARG and the ISST–TANL det/prep together have the same coverage; the same holds for the TUT AUX+PASSIVE/ AUX+TENSE relations with respect to the ISST–TANL aux relation.

5. Merging TUT and ISST–TANL

In this section we summarise the work done towards merging the two annotated resources, by defining a bridge annotation scheme to be used as an interlingua for converting the individual treebanks and combining them into a wider resource. Whereas we are aware of previous efforts of combining different annotation types (e.g. ISOTimeML, PropBank, and FrameNet annotations as reported in Ide and Bunt (2010)) as well as dependency structures of different languages (e.g. English vs Japanese as discussed in Hayashi et al. (2010)), to our knowledge this represents the first merging effort carried out with respect to different dependency annotation schemes defined for the same language: we might look at them as dependency annotation "dialects". In what follows, we first illustrate the criteria which guided the definition of a bridge annotation scheme to be used for merging the two resources (Section 5.1.); second, in order to test the adequacy of the resulting annotation scheme as far as dependency parsing is concerned we report the parsing results achieved so far by exploiting the MIDT resources as training data (Section 5.2.).

5.1. Defining a bridge annotation scheme for MIDT

The results of the comparative analysis detailed in section 3.1. are summarized in columns 2, 3 and 4 of Table 1, where for each relation type in a given scheme the corresponding relation(s) are provided as far as the other scheme is concerned. The fourth column (headed "DIFF") provides additional information for what concerns the type of correspondence holding between ISST–TANL and TUT dependency categories: two different values are foreseen, which can also be combined together, corresponding to whether the correspondence involves different head selection criteria ("Hsel") and/or a different linguistic interpretation resulting in a different coverage ("covg"). It can be noted that the emerging situation is quite heterogeneous.

The only simple cases are represented by a) the root, relative clause and passive subject cases for which we observe a 1:1 mapping, and b) the relation(s) involving auxiliaries in complex tense constructions characterized by a 1:n mapping. As far as b) is concerned, in principle the TUT relation distinctions might be recoved by also taking into account the lexical and morpho–syntactic features associated with the involved auxiliary and main verbal tokens. In both a) and b) cases, however, the identification of a bridge category to be used for merging purposes does not appear to be problematic at all (see below).

A slightly more complex case is represented by the determiner–noun, preposition–noun and complementizer–verb relations whose treatment in the two schemes is different both at the level of involved relations and head selection criteria. For these cases, the merging process should also be able to deal with the "external" consequences at the level of the overall tree structure as far as the attachment of these constructions is concerned. For instance, depending on the scheme in a sentence like *I read the book* the object of reading would be either the article (TUT) or the noun (ISST–TANL). In these cases, besides defining a semantically coherent bridge category compatible with both TUT and ISST–TANL annotations, the conversion process is not circumscribed to the dependency being converted but should also deal with the restructuring of the sub–tree whose head governs the dependency head.

Most part of remaining dependency relations involve different, sometimes orthogonal, sets of criteria for their assignment and are therefore more difficult to deal with for merging purposes. Consider, as an example, the direct object relation, already discussed in Section 3.1.: in ISST–TANL the relation obj is restricted to non–clausal objects, whereas the TUT OBJ relation also includes clausal ones. This difference in terms of coverage follows from the fact that whereas TUT implements a pure dependency annotation where the dependency type does not vary depending on the complement type (e.g. clausal vs nominal objects), in ISST–TANL all clausal complements are treated under a specific relation type, named arg. This represents a much trickier case to deal with for merging purposes: here it is not a matter of choosing between two different representation strategies, but rather of converging on a possibly underspecified representation type which could be automatically reconstructed from both TUT and ISST–TANL resources. If on the one hand in TUT it is possible to recover the ISST–TANL notion of arg by exploiting the morpho–syntactic features of the tokens involved in the relation, on the other hand it is impossible to automatically recover the TUT notion of OBJ starting from ISST–TANL annotation only (in this case information about the subcategorization properties of individual verbs would be needed).

Another problematic conversion area is concerned with the representation of deverbal nouns (e.g. *destruction*) whose annotation in TUT is carried out in terms of the underlying predicate–argument structure (i.e. by marking relations such as subject, object, etc.) whereas in ISST–TANL is marked by resorting to generic surface (e.g. comp(lement)) relations. As in the subordination case, the only possible solution here is to converge on a representation type which can be automatically reconstructed from both TUT and ISST–TANL resources by combining morfo–syntactic and dependency information.

It should also be noted that there are semantically–oriented distinctions which are part of the ISST–TANL annotation scheme (e.g. temporal and locative modifiers, i.e. mod_temp vs mod_loc) but which do not find a counterpart in the CoNLL version of the TUT treebank. In this case the only possible solution consists in neutralizing such a distinction at the level of the MIDT representation.

The conversion process had also to deal with cases for which the difference was only at the level of annotation criteria rather than of the dependency types. Consider for instance the treatment of coordination phenomena. Both TUT and ISST–TANL foresee two different relations, one for linking the conjunction to one of the conjuncts (i.e. the ISST–TANL con and the TUT COORD relations) and the other one for connecting the conjoned elements (i.e. the ISST–TANL conj and the TUT COORD2ND relations). In spite of this parallelism at the tagset level, the strategy adopted for representing coordinate structures is different in the two resources: whereas ISST–TANL takes the first conjunct as the head of the whole coordinate structure and all subsequent conjoined elements and conjunctions are attached to it, in TUT both the conjuction and the conjunct are governed by the element immediately preceding it. In this case, the conversion towards MIDT consists in restructuring the internal structure of the coordinate structure.

So far, we focused on the conversion of "canonical" dependency relations and of coordination: the treatment of punctuation is still being defined. For each set of corresponding ISST–TANL and TUT categories, the last column of Table 1 contains the MIDT counterpart. The definition of the MIDT dependency tagset was first guided by practical considerations: namely, bridge categories should be automatically reconstructed by exploiting morpho–syntactic and dependency information contained in the original ISST–TANL and TUT resources. In MIDT, we also decided to neutralize semantically–oriented distinctions (such as the subject of passive constructions, or the indirect object) which turned out to be problematic (see Section 4.) to be reliably identified in parsing in spite of their being explicitly encoded in both annotation schemes. Last but not least, the linguistic soundness of resulting categories was also assessed, by comparing the MIDT tagset with de facto dependency annotation standards: among them it is worth mentioning here the annotation tagsets proposed by the syntactic annotation initiatives like TIGER, ISST, Sparkle and EAGLES as reported in Declerck (2008) or the most recent Stanford typed dependencies representation (de Marneffe and Manning, 2008).

It should be noted that, in some cases, MIDT provides two different options, corresponding to the TUT and ISST–TANL styles for dealing with the same construction: this is the case of determiner–noun, preposition–noun, complementizer–verb and auxiliary–main verb relations whose MIDT representation is parameterizable: for the time being only one possible option has been activated. The final MIDT tagset contains 21 dependency tags (as opposed to the 72 tags of TUT and the 29 of ISST–TANL), including the different options provided for the same type of construction. The question at this point is whether the MIDT annotation scheme is informative enough and at the same time fully predictable to reliably be used for different purposes: in the following section a first preliminary answer to this question is provided.

5.2. Using MIDT as training corpus

In this section we report the results achieved by using MIDT resources for training a dependency parsing system. We used DeSR (Dependency Shift Reduce), a transition–based statistical parser (Attardi, 2006) which builds dependency trees while scanning a sentence and applying at each step a proper parsing action selected through a classifier based on a set of representative features of the current parse state. Parsing is performed bottom-up in a classical Shift/Reduce style, except that the parsing rules are special and allow parsing to be performed deterministically in a single pass. It is possible to specify, through a configuration file, the set of features to use (e.g. POS tag, lemma, morphological features) and the classification algorithm (e.g. Multi-Layer Perceptron (Attardi and Dell'Orletta, 2009), Support Vector Machine, Maximum Entropy). In addition, the parser can be configured to run either in left–to–right or right–to–left word order. An effective use of DeSR is the Reverse Revision parser (Attardi et al., 2009), a stacked parser which first runs in one direction, and then extracts hints from its output to feed another parser running in the opposite direction. All these options allow creating a number of different parser variants, all based on the same basic parsing algorithm. Further improvement can then be achieved by the technique of parser combination (Attardi et al., 2009), using a greedy algorithm, which preserves the linear complexity of the individual parsers and often outperforms other more complex algorithms.

Let us start from the results achieved by this parser in the framework of the evaluation campaign Evalita 2011 with the original TUT and ISST–TANL datasets distributed in the framework of the "Dependency Parsing" (Bosco and Mazzei, 2012) and "Domain Adaptation" (Dell'Orletta et al., 2012) tracks respectively. Table 2 reports, in the first two rows, the values of Labeled Attachment Score (LAS) obtained with respect to the ISST–TANL and TUT datasets with the technique of parser combination: 82.09% vs 89.88%. This result is in line with what reported in Bosco et al. (2010), where a similar difference in performance was observed with respect to the TUT and ISST–TANL test sets: the composition of the training corpora and the adopted annotation schemes were identified as possible causes for such a difference in performance.

The results reported in rows 3–6 have been obtained by training DeSR with the MIDT version of the TUT and ISST–TANL individual resources, whereas rows 7 and 8 refer to the merged MIDT resource. In all these cases two different LAS scores are reported, i.e. the overall score and the one computed by excluding punctuation: this was done to guarantee the comparability of results since, as pointed out above, the conversion of punctuation is still under way

Table 1: ISST–TANL, TUT and MIDT linguistic ontologies

ID	ISST–TANL	TUT	DIFF	MIDT
1	ROOT	TOP		_ROOT
2	arg	no equivalent relation (see 5, 21)	covg	_ARG
3	aux	AUX(+PASSIVE +PROGRESSIVE +TENSE)		_AUX
4	clit	EMPTYCOMPL SUBJ/SUBJ+IMPERS		_CLIT
5	comp	INDCOMPL SUBJ/INDCOMPL COORD+COMPAR	covg	_COMP
6	comp_ind	INDOBJ SUBJ/INDOBJ		_COMP
7	comp_loc	no equivalent relation(see 5)	covg	_COMP
8	comp_temp	no equivalent relation (see 5)	covg	_COMP
9	con	COORD(+BASE +ADVERS +COMPAR +COND +CORRELAT +ESPLIC +RANGE +SYMMETRIC)	covg Hsel	_COORD
10	concat	CONTIN(+LOCUT +DENOM +PREP)		_CONCAT
11	conj	COORD2ND(+BASE +ADVERS +COMPAR +COND +CORRELAT +ESPLIC) COORDANTEC+CORRELAT	covg Hsel	_COOR2ND
12	det	ARG	Hsel	_DET, _ARG
13	dis	no equivalent relation (see 9)	covg	_COORD
14	disj	no equivalent relation (see 11)	covg	_COOR2ND
15	mod	APPOSITION RMOD RMOD+RELCL+REDUC INTERJECTION COORDANTEC+COMPAR	covg	_MOD
16	mod_loc	no equivalent relation (see 15)	covg	_MOD
17	mod_rel	RMOD+RELCL		_RELCL
18	mod_temp	no equivalent relation (see 15)	covg	_MOD
19	modal	no equivalent relation (see 3)	Hsel covg	_AUX
20	neg	no equivalent relation (see 15)	covg	_NEG
21	obj	OBJ SUBJ/OBJ EXTRAOBJ	covg	_OBJ
22	pred	PREDCOMPL(+SUBJ +OBJ) RMODPRED(+OBJ +SUBJ)		_PRED
23	pred_loc	no equivalent relation (see 22)	covg	_PRED
24	pred_temp	no equivalent relation (see 22)	covg	_PRED
25	prep	ARG	Hsel	_PREP, _ARG
26	punc	CLOSE(+PARENTHETICAL +QUOTES) END INITIATOR OPEN(+PARENTHETICAL +QUOTES) SEPARATOR		_PUNC
27	sub	ARG	Hsel	_SUB, _ARG
28	subj	SUBJ EXTRASUBJ	covg	_SUBJ
29	subj_pass	OBJ/SUBJ		_SUBJ

(i.e. for the time being the original treatment of punctuation is maintained). For the MIDT resources, the DeSR results achieved with the best single parser and with the combination of parsers are reported. It can be noticed that in both cases an improvement is observed with respect to the native TUT and ISST–TANL resources, +0.23% and + 2.90% respectively. The last two rows refer to the results achieved with the merged resource used as training, which at the time of writing is far from being completed due to the fact that the treatment of punctuation has not been unified yet. In spite of this fact (which in principle could generate noise in the model), the performance achieved by training the parser on the merged resource is still high, although lower than the result achieved with TUT_MIDT_train. The parsing model trained on the merged resource obtains the following results with respect to individual test sets: 83.43% for ISST–TANL_MIDT_test and 88.03% for TUT_MIDT_test, which represent slighly lower LAS scores than those obtained by using as training the corresponding resource. In spite of the fact that the harmonization and merging of the two resources is still under way, achieved parsing results show that the resulting MIDT resource can effectively be used for training dependency parsers.

6. Conclusion

The outcome of the effort sketched in this paper is three–fold. First, a methodology for harmonizing and merging annotation schemes belonging to the same family has been defined starting from a comparative analysis carried out a) with respect to different dimensions of variation ranging from head selection criteria, dependency tagset granularity to annotation guidelines or the treatment of specific constructions, and b) by analysing the performance of state–of–the–art dependency parsers using as training the original resources. Second, Italian will have a bigger treebank, which will be further extended if other available treebanks will be involved in the merging process. Third, but not least important, the set of "bridge" categories which have been defined for merging purposes can in principle be used to enrich the set of dependency–related data categories of the ISOcat Data Category Registry, thus enabling other merging initiatives operating within the same dependency–based family of annotation schemes to start from a richer and already experimented set of basic dependency–related categories. Current directions of work include: the completion of the conversion and merging process to obtain a fully harmonised resource; the parameterizability of conversion, in order to allow for different annotation choices.

Table 2: Parsing results with native vs MIDT resources

TRAINING	TEST	PARSER	LAS	LAS no punct
ISST–TANL_native_train	ISST–TANL_native_test	Parser comb.	82.09%	not available
TUT_native_train	TUT_native_test	Parser comb.	89.88%	not available
ISST–TANL_MIDT_train	ISST–TANL_MIDT_test	Best single	84.47%	86.15%
ISST–TANL_MIDT_train	ISST–TANL_MIDT_test	Parser comb.	84.99%	86.78%
TUT_MIDT_train	TUT_MIDT_test	Best single	89.23%	90.74%
TUT_MIDT_train	TUT_MIDT_test	Parser comb.	90.11%	91.58%
merged_MIDT_train	merged_MIDT_test	Best single	86.09%	88.60%
merged_MIDT_train	merged_MIDT_test	Parser comb	86.66%	89.04%

7. Acknowledgements

The work reported in this paper was partly carried out in the framework of the national PRIN project PARLI ("Portal for the Access to the Linguistic Resources for Italian") funded by the Ministry of Public Instruction and University.

8. References

G. Attardi and F. Dell'Orletta. 2009. Reverse revision and linear tree combination for dependency parsing. In *Proceedings of NAACL HLT (2009)*.

G. Attardi, F. Dell'Orletta, M. Simi, and J. Turian. 2009. Accurate dependency parsing with a stacked multilayer perceptron. In *Proceedings of EVALITA, Evaluation of NLP and Speech Tools for Italian*, Reggio Emilia, Italy.

G. Attardi. 2006. Experiments with a multilanguage non–projective dependency parser. In *Proceedings of CoNLL'06*, New York City, New York.

C. Bosco and A. Mazzei. 2012. The evalita 2011 parsing task: the dependency track. In *Working Notes of Evalita'11*, Roma, Italy.

C. Bosco, V. Lombardo, L. Lesmo, and D. Vassallo. 2000. Building a treebank for italian: a data-driven annotation schema. In *Proceedings of LREC'00*, Athens, Greece.

C. Bosco, S. Montemagni, A. Mazzei, V. Lombardo, F. Dell'Orletta, and A. Lenci. 2009. Evalita'09 parsing task: comparing dependency parsers and treebanks. In *Proceedings of Evalita'09*, Reggio Emilia, Italy.

C. Bosco, S. Montemagni, A. Mazzei, V. Lombardo, F. Dell'Orletta, A. Lenci, L. Lesmo, G. Attardi, M. Simi, A. Lavelli, J. Hall, J. Nilsson, and J. Nivre. 2010. Comparing the influence of different treebank annotations on dependency parsing. In *Proceedings of LREC'10*, Valletta, Malta.

M. Buch-Kromann, I. Korzen, and H. Høeg Müller. 2009. Uncovering the 'lost' structure of translations with parallel treebanks. *Special Issue of Copenhagen Studies of Language*, 38:199–224.

J.C.K. Cheung and G. Penn. 2009. Topological field parsing of German. In *Proceedings of ACL-IJCNLP'09*.

M.C. de Marneffe and C.D. Manning. 2008. Stanford typed dependencies manual. Technical report, Stanford University.

T. Declerck. 2008. A framework for standardized syntactic annotation. In *Proceedings of LREC'08*, Marrakech, Morocco.

F. Dell'Orletta, S. Marchi, S. Montemagni, G. Venturi, T. Agnoloni, and E. Francesconi. 2012. Domain adaptation for dependency parsing at evalita 2011. In *Working Notes of Evalita'11*, Roma, Italy.

Y. Hayashi, T. Declerck, and C. Narawa. 2010. Laf/graf-grounded representation of dependency structures. In *Proceedings of LREC'10*, Valletta, Malta.

R. Hudson. 1984. *Word Grammar*. Basil Blackwell, Oxford and New York.

N. Ide and H. Bunt. 2010. Anatomy of annotation schemes: mapping to graf. In *Proceedings of the Linguistic Annotation Workshop*, Uppsala, Sweden.

N. Ide and L. Romary. 2006. Representing linguistic corpora and their annotations. In *Proceedings of LREC'06*, Genova, Italy.

N. Ide and K. Suderman. 2007. GrAF: A graph-based format for linguistic annotations. In *Proceedings of the Linguistic Annotation Workshop*, Prague, Czech Republic.

M. Kemps-Snijders, M. Windhouwer, P. Wittenburg, and S.E. Wright. 2009. Isocat: remodelling metadata for language resources. *IJMSO*, 4(4):261–276.

S. Kübler, R.T. McDonald, and J. Nivre. 2009. *Dependency Parsing*. Morgan & Claypool Publishers, Oxford and New York.

G. Leech, R. Barnett, and P. Kahrel. 1996. Eagles recommendations for the syntactic annotation of corpora. Technical report, EAG-TCWG-SASG1.8.

A. Lenci, S. Montemagni, V. Pirrelli, and C. Soria. 2008. A syntactic meta–scheme for corpus annotation and parsing evaluation. In *Proceedings of LREC'00*, Athens, Greece.

S. Montemagni and M. Simi. 2007. The Italian dependency annotated corpus developed for the CoNLL–2007 shared task. Technical report, ILC–CNR.

S. Montemagni, F. Barsotti, M. Battista, N. Calzolari, O. Corazzari, A. Lenci, A. Zampolli, F. Fanciulli, M. Massetani, R. Raffaelli, R. Basili, M. T. Pazienza, D. Saracino, F. Zanzotto, N. Mana, F. Pianesi, and R. Delmonte. 2003. Building the Italian Syntactic-Semantic Treebank. In A. Abeillé, editor, *Building and Using syntactically annotated corpora*. Kluwer.

J. Nivre, J. Hall, S. Kübler, R. McDonald, J. Nilsson, S. Riedel, and D. Yuret. 2007. The CoNLL 2007 shared task on dependency parsing. In *Proceedings of EMNLP-CoNLL'07*.

S. Tonelli, R. Delmonte, and A. Bristot. 2008. Enriching the venice italian treebank with dependency and grammatical relations. In *Proceedings of LREC'08*, Marrakech, Morocco.