The *Index Thomisticus* Treebank Project: Annotation, Parsing and Valency Lexicon

Barbara McGillivray*, Marco Passarotti, Paolo Ruffolo****

** University of Pisa, Italy*
b.mcgillivray@ling.unipi.it
*** Catholic University of the Sacred Heart, Milan, Italy*
marco.passarotti@unicatt.it, paolo.ruffolo@poste.it

ABSTRACT. *We present an overview of the Index Thomisticus Treebank project (IT-TB). The IT-TB consists of around 60,000 tokens from the Index Thomisticus by Roberto Busa SJ, an 11-million-token Latin corpus of the texts by Thomas Aquinas. We briefly describe the annotation guidelines, shared with the Latin Dependency Treebank (LDT). The application of data-driven dependency parsers on IT-TB and LDT data is reported on. We present training and parsing results on several datasets and provide evaluation of learning algorithms and techniques. Furthermore, we introduce the IT-TB valency lexicon extracted from the treebank. We report on quantitative data of the lexicon and provide some statistical measures on subcategorisation structures.*

RÉSUMÉ. *Nous présentons une vue d'ensemble du projet de l'Index Thomisticus Treebank (IT-TB). L'IT-TB consiste d'environ 60,000 occurrences tirées de l'Index Thomisticus de Roberto Busa SJ, un corpus de onze millions de mots latins de Thomas d'Aquin. Nous décrivons brièvement les règles d'étiquetage, qui sont en commun avec la Latin Dependency Treebank (LDT). Nous décrivons l'application des parseurs probabilistes dépendanciels sur les données de l'IT-TB et de la LDT. Nous présentons les résultats de l'entraînement et de l'analyse syntactique sur plusieurs ensembles des données et nous fournissons une évaluation des algorithmes et des techniques d'apprentissage. En outre, nous introduisons le lexique de valence de l'IT-TB tiré de la treebank. Nous reportons les données quantitatives du lexique et nous fournissons quelques mesures statistiques sur les structures de sous-catégorisation.*

KEYWORDS: *treebanks, Latin, parsing, valency lexicon*

MOTS-CLÉS : *treebanks, latin, analyse syntactique automatique, lexique de valence*

1. Introduction

The *Index Thomisticus* (IT) (Busa, 1974-1980) was begun in 1949 and is considered to be a groundbreaking project in computational linguistics. It contains the *Opera omnia* of Thomas Aquinas in digital form (118 texts) as well as 61 texts by other authors related to Aquinas, for a total of around 11 million tokens. The corpus is morphologically tagged and lemmatised. Early in the 1970s Busa started to plan a project aimed at both the morphosyntactic disambiguation of the IT lemmatisation and the syntactic annotation of its sentences.

Today, both these tasks are being undertaken by the "*Index Thomisticus* Treebank" project (IT-TB), [1] which is part of the wider "Lessico Tomistico Biculturale" (LTB), whose goal is the development of a lexicon on the basis of the IT texts.

This paper describes some of the main achievements of the IT-TB project, paying particular attention to aspects applicable to natural language processing (NLP). The paper is organised as follows: section 2 describes the available Latin language resources and the state of the art of NLP for Latin; section 3 deals with structural features of Latin syntax and provides information on the available Latin treebanks and their annotation guidelines; section 4 describes the IT-TB project in more detail, describing its annotation procedures, its use and evaluation of parsers, and the IT-TB valency lexicon; finally, some research perspectives and concluding remarks are sketched in section 5.

2. Language resources and NLP tools for Latin

Despite its pioneering role in computational linguistics, due in particular to the IT itself, today Latin still lacks powerful NLP tools that can automatically process layers of annotation higher than the morphological layer, nor are there state-of-the-art language resources such as annotated corpora and lexical databases. Indeed, only a few of the huge number of Latin texts currently available in digital format have been even morphologically tagged, while most of them are not linguistically tagged at all. During recent decades, several research centres and projects have digitised many Latin texts. There are, for instance, the large databases provided by CTLO (*Centre "Traditio Litterarum Occidentalium"* [2]) and by LASLA at the University of Liège (*Laboratoire d'Analyse Statistique des Langues Anciennes*; (Denooz, 1996)), or the Perseus Digital Library (Crane *et al.*, 2001) at Tufts University in Boston. In particular, LASLA has produced a 1.5-million-word database which is morphosyntactically lemmatised and syntactically annotated at the clausal level. Verbs for main

1. IT-TB data can be browsed online through the searcher and viewer Netgraph (Mírovský, 2006) at the following url: http://itreebank.marginalia.it.

2. The CTLO, directed by Paul Tombeur, is located in Turnhout (Belgium) and since 2001 has continued the activities formerly carried out by CETEDOC (*CEntre de Traitement Electronique des DOCuments*) which was located at the Catholic University of Louvain-la-Neuve (Belgium).

and subordinated clauses are distinguished, the latter being tagged as ablative absolute or as accusative plus infinitive. In regards to lexical databases, in addition to the traditional Latin dictionaries and lexica available on-line or on CD-ROM, such as Lewis-Short provided by the Perseus Digital Library website or the *Thesaurus Linguae Latinae* from the Bayerische Akademie der Wissenschaften in Munich, it is also worth mentioning the *Thesaurus Formarum* (TF-CILF) from the CTLO and the *Neulateinische Wortliste* made available by Johann Ramminger (http://www.lrz-muenchen.de/~ramminger/). A strong advance in this field will come from the ongoing development of Latin WordNet, which is integrated within the existing MultiWordNet project (http://multiwordnet.itc.it/english/home.php) aimed at the realisation of a large-scale multilingual computational lexicon based on WordNet. WordNet is a lexicon-oriented semantic network, started at Princeton University for the English language, in which lexical items are organised into sets of synonyms ("synsets"), representing lexical concepts. WordNets for many other languages have been created so far and their synsets are linked with each other on the basis of the Princeton synsets. The links among the synsets are defined by means of semantic and lexical relations; semantic relations, such as hypo/hyperonymy and meronymy, hold among synsets, while lexical relations, such as antonymy, among words. Presently, the size of Latin WordNet is around 10,000 lemmas, 9,000 synsets and 25,000 word senses. Although WordNet represents a powerful language resource for NLP tasks such as information extraction, data mining, word sense disambiguation and topic classification, the most advanced NLP tools for Latin are still far from the automatic processing of such 'semantic' tasks.

Three morphological Latin analysers are nowadays available. They are CHLT-LEMLAT (Passarotti, 2007a), Whitaker's *Words* and *Morpheus* (Crane, 1991), this latter being first developed in the Perseus Digital Library for Ancient Greek in 1985 and extended to support Latin in 1996. In the field of automatic processing of Latin morphology, (Schinke *et al.*, 1998) report on a system for the retrieval of inflectional variants in Latin databases: this is not a morphological analyser, but more a retrieval system allowing users to carry out searches on textual databases.

Specific tools for morphosyntactic disambiguation and Part-of-Speech tagging have been developed by LASLA for the annotation of their textual database. As far as parsing is concerned, a first attempt at Latin dependency parsing is described in (Koch, 1993), who reports on the enhancement for Latin of an existing dependency parser (Covington, 1990). (Koster, 2005) describes a rule-based top-down chart parser, automatically generated, developed from a grammar, and a lexicon built according to the formalism of the two-level AGFL (Affix Grammar over a Finite Lattice; (Koster, 1991)) grammar. Recently, a hybridisation of this parser has been developed, extending the rule-based core parser with a probability-based ranking of dependency trees, through statistics of dependency triplets generated by the parser itself. A real opportunity for advancing the state of the art of Latin linguistic resources and NLP has been provided by the start, in 2005, of two projects aimed at developing syntactically annotated corpora for Latin (treebanks). These are the *Index Thomisticus* Treebank by the Catholic University of the Sacred Heart in Milan on texts from the IT

(Passarotti, 2007b) and the Latin Dependency Treebank (LDT) by the Perseus Digital Library in Boston on texts of the Classical era (Bamman and Crane, 2007). Later on, a third Latin treebank was started at the University of Oslo, as part of the wider project PROIEL (*Pragmatic Resources in Old Indo-European Languages*) aimed at the syntactic annotation of the oldest extant versions of the New Testament in Indo-European languages: Latin, Greek, Gothic, Armenian and Old Church Slavonic (Haug and Jøndal, 2008). Recently, another project, named CoLaMer, collaboratively developed by the Universities of Köln and Regensburg, started to develop a fourth Latin treebank on texts of the Merovingian age.

The size of these treebanks is presently around 60,000 annotated tokens for the IT-TB, 60,000 for the LDT and 100,000 for the Latin section of the PROIEL treebank, while CoLaMer is now starting the annotation phase.

The syntactically annotated data provided by these treebanks can be used as training sets for Part-of-Speech taggers and probabilistic dependency parsers, in order to reach accuracy rates in the NLP of Latin texts which are similar to those available for other less resourced languages.

3. Latin treebanks

3.1. *Features of Latin syntax*

Latin is a richly inflected language, showing the following syntactic features:

– discontinuous phrases ("non-projectivity"): this means that phrases may be not continuous, but broken up by words of other phrases. An example is the following sentence by Ovid (*Metamorphoses*, I.1-2): "In nova fert animus mutatas dicere formas corpora" ("My mind leads me to tell of forms changed into new bodies"). In this sentence, both the nominal phrases "nova corpora" and "mutatas formas" are discontinuous;

– moderately free word-order: for instance, the order of the words in a sentence like "audentes fortuna iuvat" ("fortune favours the bold"; Vergil, *Aeneid*, X, 284) could be changed into "fortuna audentes iuvat", or "fortuna iuvat audentes", without affecting the meaning of the sentence.

These features of Latin syntax influenced the choice of Dependency Grammars (DG) as the most suitable grammar framework for building Latin treebanks. While in the 1970s the first treebanks were annotated via Phrase Structure Grammar (PSG)-based schemata (as in the IBM, Lancaster and, later on, Penn treebanks), over the past decade many treebank projects have been undertaken using DG, such as the ALPINO treebank for Dutch (Van der Beek *et al.*, 2002), the Prague Dependency Treebank (PDT) for Czech (Böhmová *et al.*, 2003), and the Danish Dependency Treebank (Kromann, 2003). The reason behind this is that the first treebanks were mainly English-language corpora. PSG's were a suitable framework for the syntactic description of a poorly inflected language like English, showing fixed word-order and

few discontinuous constituents. Later on, the syntactic annotation of moderately free word-order languages (like Latin) required the adoption of the DG framework, which is more appropriate than PSG for such a task. Furthermore, (Carroll *et al.*, 1998a) showed that inter-annotator agreement was significantly better for dependency treebanks, indicating that PSG annotation was requiring too many irrelevant decisions (see also (Lin, 1995)).

3.2. *The annotation guidelines for Latin treebanks*

Sharing the DG framework of annotation, the IT-TB and LDT have worked collaboratively since the beginning of their respective projects (Bamman *et al.*, 2007b). Since the IT-TB and LDT are the first projects of their kind for Latin, no prior established guidelines were available to rely on for syntactic annotation. So the decision was made to follow the PDT guidelines for the so-called "analytical layer" of annotation (Hajič *et al.*, 1999), which was adapted for the treatment of specific or idiosyncratic constructions of Latin that could be syntactically annotated in several different ways. These constructions (such as the ablative absolute or the passive periphrastic) are common to Latin of all eras. Rather than have each treebank project decide upon and record each decision for annotating them, IT-TB and LDT decided to pool their resources and create a single annotation manual that would govern both treebanks ((Bamman *et al.*, 2007a), (Bamman *et al.*, 2007b)). When dealing with Latin dialects separated by 13 centuries, sharing a single annotation manual is very useful for comparison purposes, such as checking annotation consistency or diachronically studying specific syntactic constructions (Bamman *et al.*, 2008b). In addition, the task of data annotation through these common guidelines allows annotators to base their decisions on a variety of examples from a wider range of texts and combine the two datasets in order to train probabilistic dependency parsers. Table 1 lists all of the syntactic tags currently in use in IT-TB and LDT (Bamman *et al.*, 2008a).

As in the PDT, all of the tags can be appended with a suffix in the event that the given node is a member of a coordinated construction (_Co), an apposition (_Ap) or a parenthetical statement (_Pa). The tag Pred is given to the predicate of the main clause (or clauses, in case of coordination or apposition) of a sentence; the head verbs of the subordinate clauses are annotated according to the role of the clause in the sentence (for instance, a declarative clause acting as subject is annotated with the tag Sb). An Atr is a sentence member that further specifies a noun in some respect; typical attributives are adjectives (*bonus puer*, "good boy") and nouns in the genitive case (*domus patris*, "the father's house"). The difference between Obj and Adv roughly corresponds to that between arguments (inner participants) and adjuncts of verbs or adjectives, i.e., between those called "actants" and "circonstants" in the terminology of (Tesnière, 1959). A special kind of Obj is the determining complement of the object, which is tagged with OComp, such as *senatorem* in a sentence like "aliquem senatorem facere" ("to nominate someone senator"). The determining complement of the subject is, conversely, tagged using PNom; this mainly occurs in case of construc-

Pred	predicate
Sb	subject
Obj	object
Atr	attributive
Adv	adverbial
Atv/AtvV complement	complement
PNom	predicate nominal
OComp	object complement
Coord	coordinator
Apos	apposing element
AuxP	preposition
AuxC	conjunction
AuxR	reflexive passive
AuxV	auxiliary verb
AuxX	commas
AuxG	bracketing punctuation
AuxK	terminal punctuation
AuxY	sentence adverbials
AuxZ	emphasising particles
AuxS	root of the tree
ExD	ellipsis

Table 1. *Complete Latin tagset*

tions like “aliquis senator fit” (“someone becomes a senator”). The tag OComp covers some of the functions of the Atv/AtvV tag (Verbal Attribute) as used by the PDT. However, departing from the PDT style, we assign a different tag to object complements (OComp) and to complements that are not direct arguments of the verb (Atv/AtvV). These are usually noun phrases and adjectives that agree with their head noun morphologically, but differ from typical attributes in that they also qualify the function of the verb; the use of Atv/AtvV is largely similar to the account of “praedicativa” given in (Pinkster, 1990): 142-162. The CoLaMer project follows the same annotation style developed for the IT-TB and LDT. While the PROIEL annotation guidelines are grounded on the same grammar framework as the IT-TB and LDT, they differ in a number of details, some of which are described below. A conversion phase is now ongoing, in order to have the four treebanks annotated in the same way in the near future. PROIEL makes use of several more specific tags than the IT-TB and LDT, such as NARG for the arguments of nouns (for instance, *in sanctitatem* in “ingressio in sanctitatem Dei”, “entrance in God’s sanctity” [3]), AG(ent), covering agents in

3. The examples are excerpted from PROIEL guidelines, which are available on-line at http://www.hf.uio.no/ifikk/proiel/publications/guidelines.pdf.

passive constructions (*ab his* in "quae tibi obiciuntur ab his", "[those things] that are objected to you by these") and OBL(ique), assigned to those verbal arguments that are not subject or object to the clausal node (*mihi* in "et dixit mihi angelus", "and the angel told me"). OBL includes also non-accusative objects (such as the ablative object of *utor*, "to use") as well as prepositional arguments (*eum* in 'et introibo ad eum", "and I will enter him"). Another difference is that AuxC and AuxP tags are not adopted in PROIEL: conjunctions and prepositions are tagged according to the sentence role of the phrase, or the subordinate clause they introduce. Conversely, in the LDT and IT-TB style, this annotation is given to the main predicate of the clause introduced by the conjunction, or to the prepositional argument(s): as in the PDT, conjunctions and prepositions are considered as "bridge" auxiliary structures (respectively tagged with AuxC and AuxP). For instance, in the tree of the sentence "cenabo cum illo" ("I will have dinner with him"), *illo* depends on *cum*: in such a tree, PROIEL assigns Adv to *cum* and OBL to *illo*, while in LDT and IT-TB *cum* is an AuxP and *illo* is an Adv.

4. The *Index Thomisticus* Treebank

4.1. *Annotation procedures*

Up to now, the annotation of IT-TB data has been performed both manually and semi-automatically, using the tree editor TrEd, developed by Petr Pajas for the PDT. [4] The annotation of a sentence requires the three following steps:

1) checking and (possibly) correcting the IT morphological analysis. The texts contained in the IT are tagged such that among the possible morphological analyses of each word, only the first possible option in the grammars is assigned. For instance, a word like *puella* is always tagged as a singular nominative and never as a singular vocative, or ablative;

2) assigning to each word a syntactic tag;

3) defining and designing the relations between the words in the tree.

Figure 1 shows a sentence tree before manual annotation is performed. The sentence is shown in the upper part of the screen: "contra, Boetius dicit quod simplex forma subjectum esse non potest" ("on the other hand, Boethius says that a simple form cannot be subject"). [5] The tree is shown below. Each node in the tree corresponds to a word in the sentence (and vice versa), except for the root, which indicates the number of the sentence in the treebank (in this case, it is the first one); as Figure 1 shows, before the annotation is performed all the nodes are linked to the root. In the lower part, the morphological tagging of the selected word in the sentence (in Figure 1, *simplex*) is highlighted. The word *simplex* is morphologically tagged as a form of the lemma *simplex*, with the following morphological tags: nominal-adjectival inflection

4. TrEd is freely available at http://ufal.mff.cuni.cz/ pajas/tred/.

5. Thomas, *Super Sententiis Petri Lombardi*, I, Quaestio 1, Articulus 4, Argumentum 1, 6-1, 7-8. The edition of the text recorded in the IT is (Aquinas, 1856-1858).

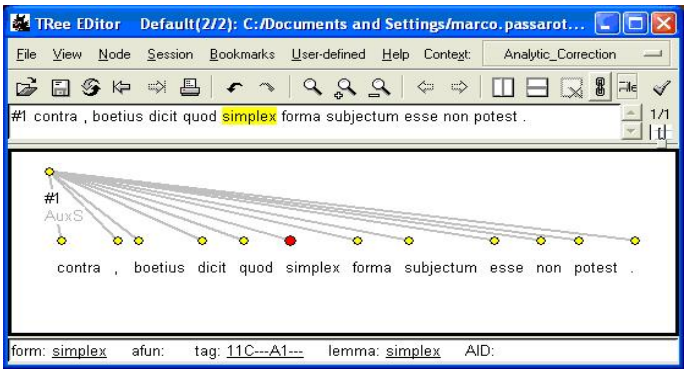

Figure 1. *Pre-annotation sentence*

(1), non-comparative degree (1), third declension noun - second class adjective (C), singular nominative (A), masculine (1). Figure 2 shows the tree of the sentence after manual annotation. In Figure 2, each node of the tree is annotated with a syntactic

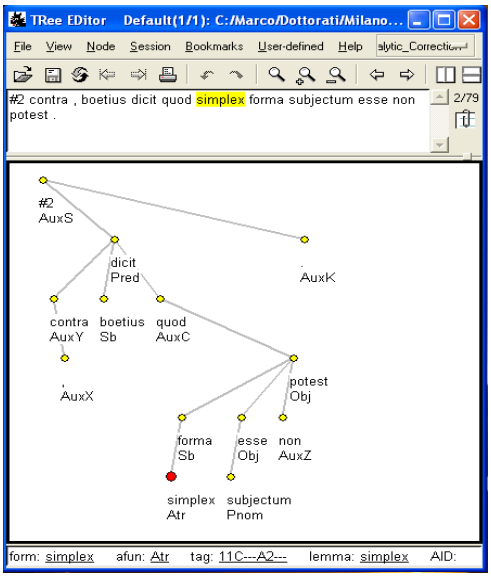

Figure 2. *Post-annotation sentence*

tag (for instance, the node of the word *forma* is tagged with “Sb” - “Subject”); the syntactic relations are represented by the branches of the tree, and the correction of the morphological tagging has been performed: *simplex* is now correctly annotated

with the feminine gender (tag "2" in the eighth position), instead of the masculine ("1").

4.2. *Parsing procedures*

The first phase of the IT-TB and LDT projects consisted primarily in the manual annotation of data. The guidelines for annotation, designed before starting the annotation task, have been tested and modified thanks to the annotation itself, which has been performed on texts separated by a wide time period (from Cicero to Aquinas), covering a great range of different styles.

In order to increase the efficiency of the annotators, both in terms of quality and of speed, the IT-TB project trained and tested a number of different probabilistic dependency parsers, exploiting the available annotated data, in order to use the best scoring parser to annotate the IT data. This allowed a semi-automatic annotation, so that annotators no longer have to draw trees from scratch; rather, starting with trees produced by the parser, they check the correctness of the analysis produced by the parser and manually eliminate the mistakes.

The sections below provide a description of the trained and tested parsers, the data sets and a discussion of the results.

Several experiments were performed with several data sets and parsers in order to explore their potential and their behaviours in processing the Latin language.

4.2.1. *Data Description*

The data used in the following experiment were taken from the publicly-available databases of the IT-TB and the LDT: the former are in CSTS format (Czech Sentence Tree Structure) and the latter in XML. The original data were converted to CoNLL format (Computational Natural Language Learning): in order to make the two sources more homogeneous, the LDT morphological tagset was converted to IT format by means of a simple mapping. Every word is associated to the correct PoS tag. Each data set used in the training/parsing experiments was randomly partitioned into a training set and a data set, so that the number of sentences in the former were in a ratio of 9:1 with the latter. The original LDT partitioning into distinct corpora by author was also used in order to perform parsing tests on dissimilar samples. Table 2 reports the size of datasets in terms of number of sentences and tokens.

According to the recommendations of the developers of the parsers, the implied training sets are close to the minimum required size, but still enough to perform significant experiments. The randomly chosen test sets should be a good criterion to evaluate and validate the quality of each trained parser.

We calculated the number of non-projective tokens in data sets: that is, the number of tokens for which, between the head and its dependent node (in left-right order), there can only be direct or indirect dependence of the head. The complexity of the

Data Sets	Sentences	Tokens
IT-Train	2007	44195
IT-Test	243	5697
LDT-Train	3093	47662
LDT-Test	380	5481
Caesar	71	1488
Cicero	327	6229
Jerome	405	8382
Ovid	316	4789
Petronius	1114	12474
Propertius	361	4857
Sallustius	701	12311
Vergil	178	2613

Table 2. *Data Sets*

LDT structure is reflected in a high rate of non-projectivity (Nivre and Nilsson, 2005) (table 3).

Data Sets	Tokens	Non-Projective	Rate
IT-Train	44195	1435	3.25%
IT-Test	5697	181	3.18%
IT	49892	1616	3.24%
LDT-Train	47662	3194	6.70%
LDT-Test	5481	339	6.19%
LDT	53143	3533	6.65%

Table 3. *Non-projectivity tokens in Data Sets (percentage of total tokens)*

4.2.2. *Parsers Description*

The parsers we decided to use were taken from the top-ranking list of the CoNLL-X Shared Task. In particular we chose the following parsers: DeSR [6] (Attardi, 2006; Attardi and Ciaramita, 2007; Attardi *et al.*, 2007), MaltParser [7] (Nivre and Scholz,

6. http://desr.sourceforge.net/.
7. http://maltparser.org/index.html.

2004; Nivre and Nilsson, 2005), MST [8] (McDonald and Pereira, 2006; McDonald *et al.*, 2005), ISBN [9] (Titov and Henderson, 2007).

All the parsers are free and open source, well documented and easy to install. MaltParser and MST parser were developed in Java, so they are completely platform independent. DeSR and ISBN (or IDP) are developed in C/C++ so they should be portable too: in our experiments we installed the latest release of each application on a GNU/Linux platform.

DeSR, MaltParser and ISBN implement Shift-Reduce Parsing algorithms, while MST implements Minimum-Spanning Tree techniques. The Shift-Reduce technique allows good computational performance, but it could be less accurate than other procedures based on global optimal criteria like the Minimun-Spanning Tree, in the case of complex structures (e.g. non-projectivity).

DeSR and MaltParser use LIBSVM (a freely available implementation of the Support Vector Machine classification algorithm) to perform the learning process. [10] ISBN implements a specific training algorithm based on Bayesian networks.

4.2.3. *Parsers Parameters Variation*

All the parsers (particularly MaltParser) are highly customisable and may be adapted to specific features of the input language. We performed many experiments on IT data in order to get the best tuning of the main parameters of each parser. As a quality measure we adopted the de facto standards: Labeled Attachment Score (LAS), Unlabeled Attachment Score (UAS) and Label Accuracy (LA)(Buchholz and Marsi, 2006). [11]

4.2.3.1. Algorithms Selection

Both MaltParser and MST allow the selection of the parsing algorithm. MaltParser implements:

– Nivre's algorithm ((Nivre, 2003), (Nivre, 2004)): a linear-time algorithm limited to projective dependency structures. Two modes are available: arc-eager or arc-standard;

– Covington's algorithm (Covington, 2001), a quadratic-time algorithm for unrestricted dependency structures. It can be run in projective mode or non-projective mode.

8. http://www.seas.upenn.edu/ strctlrn/MSTParser/MSTParser.html.

9. http://cui.unige.ch/-titov/idp/.

10. DeSR actually allows the use of several learning algorithms (Averaged Perceptron, Maximum Entropy, memory-based learning using TiMBL).

11. LAS is the percentage of tokens with correct head and relation label; UAS is the percentage of tokens with correct head; LA is the percentage of tokens with correct relation label.

MST offers a two-mode algorithm, one for non-projective data structures and one for projective data structures. We tested all such algorithms in their default settings, training MST and MaltParser on our main training set (IT-Train) and parsing the IT-Test test set. We found that Covington's algorithm slightly outperforms Nivre's arc-eager (64.72% against 63.79% of LAS), but pays a considerable computational cost (fifteen minutes against three hours on a generally available machine), while arc-standard performance is considerably lower (57.90% of LAS). As for MST, the non-projective mode achieves better accuracy than the projective one (68.79% against 67.15% of LAS).

4.2.3.2. Parsing with different features sets

Shift-Reduce parsers depend heavily on the selection of the feature sets necessary for the computational process. Choosing the correct features requires in-depth investigation. For the current experiments we simply selected, among the features sets used for other languages in the CoNLL-X Shared Task [12], the one that fits best. In particular we tested the features sets for English (default choice), Italian and Czech: both the features sets for Italian and the one for Czech are considerably better than the one for English (an average increment of about 4.82% for LAS, UAS and LA was obtained for MaltParser output).

4.2.3.3. Non-Projectivity Effects on Parsing Algorithms

The MaltParser non-projectivity algorithms were further improved by setting an ad-hoc parameter (pproj) that controls the pseudo-projectivity transformation of the input data (Nivre and Nilsson, 2005). In our experiments a slight improvement was achieved for all the possible values of the parameter (Baseline, Head, Path, Head+path [13]). Baseline setting performed the best.

4.2.4. *Best Parsers Results*

The best results achieved by each parser on IT data are summarised on table 4. For all Shift-Reduce parsers (MaltParser, DeSR and ISBN) the features set for Italian were used. DeSR and ISBN parsers have been run in their default configuration. For MaltParser the arc-eager parsing algorithm and the Baseline pseudo-projective algorithm were selected. MST has been set up for the non-projective mode.

4.2.5. *Test Sets and Training Sets Variation*

We used DeSR to perform the following three cross-treebanks experiments:

1) first of all, LDT data were parsed using DeSR trained on IT-TB data;

2) then the complementary experiment (i.e. using DeSR trained on the LDT to parse the IT-TB training set) was carried out;

12. For features sets details see: http://w3.msi.vxu.se/users/jha/conll07/.
13. See (Nivre and Nilsson, 2005).

Parser	Model descr.	LAS	UAS	LA
DeSR	default	71.26%	78.35%	81.07%
Malt	Nivre arc-eager	69.85%	75.87%	81.74%
MST	non-projective	68.79%	79.43%	79.35%
ISBN	default	68.97%	77.79%	78.88%

Table 4. *Best accuracy results for each parser (training set: IT-Train, test set: IT-Test)*

3) finally, a training set was built as a union of both the treebanks data.

Training Set	Test Set	LAS	UAS	LA
IT-Train+LDT-Train	LDT-Test	50.44%	59.52%	63.78%
LDT-Train	LDT-Test	51.18%	60.28%	63.67%
IT-Train	IT-Test	71.26%	78.35%	81.07%
IT-Train+LDT-Train	IT-Test	71.82%	78.59%	81.89%
LDT-Train	IT-Test	12.98%	30.74%	19.30%
IT-Train	Caesar	10.70%	18.22%	14.53%
IT-Train	Cicero	12.12%	18.80%	16.15%
IT-Train	Jerome	12.91%	19.32%	14.96%
IT-Train	Ovid	9.16%	18.04%	14.10%
IT-Train	Petronius	12.96%	24.73%	15.57%
IT-Train	Propertius	8.07%	17.91%	13.07%
IT-Train	Sallustius	11.01%	19.80%	14.83%
IT-Train	Vergil	9.43%	19.39%	12.42%

Table 5. *Cross-parsing accuracy results*

Table 5 reports the accuracy rates for such experiments. The LAS accuracy rates resulting from the first two experiments are very low, being an average of 10.79% where IT data are used as training set and LDT as test set, and 12.98% in the complementary experiment. Although both the treebanks adopt the same annotation guidelines, it seems that the dissimilarity between the syntax of the texts in the IT-TB and LDT datasets is so high that the data from one treebank cannot be used to train parsers to be applied on the other treebank data. In particular, as far as the second experiment is concerned, we report in table 5 the accuracy rates of the LDT texts by author. This emphasises that the lowest performances are achieved where poetry texts (by Ovid, Propertius and Vergil) are considered. The results of the third experiment do not show an improving accuracy on LDT-Test (decreasing from 51.18% to 50.44%), while on IT-Test the accuracy is slightly better (increasing from 71.26% to 71.82%). Finally,

we observe that the low parsing accuracy of the LDT data where DeSR is trained on the LDT itself (51.18%) is consistent with the results reported by (Bamman and Crane, 2008).

4.2.6. *In-Depth Results Evaluation*

Using MaltEval grouping features and its configurable evaluation settings (Nilsson and Nivre, 2008), we were able to perform a continuously-running data analysis. Grouping the accuracy performance by dependency relation (Deprel; table 6), we were able to explore in depth the behaviour of the parsers.

Deprel	DeSR	Malt	ISBN	MST
Adv	74.78%	72.63%	67.93%	65.32%
Adv_Co	51.02%	43.56%	42.62%	31.96%
Atr	79.59%	79.66%	78.36%	81.75%
AuxC	65.00%	68.35%	67.73%	74.40%
AuxK	97.46%	100.00%	99.57%	100.00%
AuxP	78.68%	73.86%	76.48%	76.99%
AuxX	80.71%	71.74%	81.21%	81.32%
AuxY	76.92%	66.44%	64.23%	70.59%
AuxZ	73.54%	72.45%	70.97%	68.18%
Coord	56.51%	58.43%	49.38%	57.88%
ExD	74.57%	79.59%	79.17%	68.45%
Obj	79.11%	79.06%	77.64%	72.79%
PNom	72.79%	69.13%	71.13%	71.71%
Pred_Co	57.94%	57.48%	58.33%	47.74%
Sb	76.26%	75.93%	77.87%	72.33%

Table 6. *Accuracy Results Grouped by Deprel (only Deprels with a frequency higher than 2% are reported)*

As table 7 shows, the accuracy rate of the main dependency relations (Adv, Atr, Obj, PNom, Sb) is pretty high, being over 70% [14]. A high accuracy rate is achieved also for the ellipsis relation (ExD), Malt and ISBN parsers reaching almost 80%. On the contrary, the lowest rates are attained where coordination is concerned: Adv_Co, Pred_Co and Coord relations are all correctly tagged in less than 60% of cases. The different parsers show quite similar accuracy rates on the single dependency relations, the main differences being between the Shift-Reduce parsers and MST. In particular, Shift-Reduce parsers perform better on ExD, Obj and Pred_Co relations, while MST reaches a higher accuracy rate on AuxC. As shown, DeSR is the best perform-

14. The dependency relation Pred is not reported in table 7, its frequence in IT-Test being lower than 2%: this is not surprising, since no more than one Pred relation is allowed in one tree.

ing parser: it behaves remarkably better than the others on Adv, Adv_Co and AuxY relations.

4.3. *The IT-TB Valency Lexicon*

4.3.1. *Motivation*

By valency of a verb we mean the number and type of complements it requires ("arguments"), as opposed to non-obligatory complements ("adjuncts") [15]. For example, the Latin verb *do* ("to give") typically displays three arguments in an active clause – a nominative subject, an accusative direct object and a dative indirect object – as in the following sentence [16], which contains the arguments *dominus*, *discipulis* and *formam*:

dominus	*discipulis*	*formam*	*baptizandi*	*dedit*
Lord-NOM.M.SG	disciples-DAT.M.PL	form-ACC.F.SG	baptize-GERUND-GEN	give-PRF.3SG

"the Lord gave to the disciples the form of the baptism" [1]

Valency lexicons for verbs usually record the arguments required by each verbal entry, called "subcategorisation frames" or "valency frames". In the previous example, the valency frame for the verb *do* is Subject_nominative+Object_dative+Object_accusative. These lexicons prove to be very useful in several NLP applications, such as parsing, word sense disambiguation, automatic verb classification and selectional preference acquisition (see, for example, (Carroll *et al.*, 1998b)). Valency lexicons can also support the creation of treebanks with regard to consistency of annotation, since they provide annotators with essential information about the number and types of verbal arguments realised at the syntactic level, along with semantic information on lexical preferences (Urešová, 2004). In recent years, several valency lexicons have been built within different theoretical frameworks. Some of these resources were created in an intuition-based fashion, and then supported by examples from corpora (see, for instance, PDT-Vallex, (Hajič *et al.*, 2003)). In addition to such intuition-based resources that were manually built, a number of valency lexicons have been automatically acquired from annotated corpora, such as VALEX (Korhonen *et al.*, 2006) for English and LexShem (Messiant *et al.*, 2008) for French. Unlike man-made lexicons, these corpus-driven resources aim at systematically reflecting the evidence of the corpus they were extracted from, and are not prone to human errors such as omissions and inconsistencies. In addition, such lexicons are able to display statistical information such as the observed frequency of subcategorisation frames as attested in the original corpora. Finally, they are less costly than hand-crafted lexical resources in terms of time, money and human resources. While several subcategorisation lexicons have been compiled for

15. See (Bühler, 1934), (Tesnière, 1959).

16. Thomas, *Super Sententiis Petri Lombardi*, IV, Distinctio 8, Quaestio 1, Articulus 3C, Argumentum 2, 3-3.4-1.

modern languages, much work in this field is still needed for classical languages such as Greek and Latin.

Regarding Latin, (Happ, 1976) reports a manually compiled list of Latin verbs, along with their valencies and some quantitative information extracted from a sample corpus of 800 verbal occurrences in Cicero's *Orationes*. More recently, (Bamman and Crane, 2008) describe a "dynamic lexicon" automatically extracted from the Perseus Digital Library, using the LDT as a training set. This lexicon displays qualitative and quantitative information on subcategorisation patterns and selectional preferences of each word as it is used in every Latin author of the corpus. Their procedure reduces the noise caused by the automatic pre-processing of the data (morphological tagging and statistical syntactic parsing), by extracting only the most common arguments and their most common lexical fillers. We introduce here a corpus-driven valency lexicon for Latin verbs (McGillivray and Passarotti, 2009). This lexicon was automatically extracted from IT-TB data using MySQL database queries; due to this automatic extraction, the lexicon is updated as the treebank size increases. The procedure can also be extended to LDT data, thanks to the common annotation guidelines.

4.3.2. *The subcategorisation structures in the valency lexicon*

In this section we describe the formal representation of the lexicon both by defining the subcategorisation structures [17] making up its entries and by illustrating them through examples from data. Each verbal entry of the lexicon is provided with all the subcategorisation structures it occurs with in the treebank. Since these structures record the verbal arguments of the verb, they reflect the information on the verb's valency we referred to in the previous section. The subcategorisation structures consist of those nodes labelled with the tags assigned to the arguments: Sb (Subject), Obj (Object), OComp (Object Complement) and PNom (Predicate Nominal). We distinguished two main structures – subcategorisation frames (SCFs) and subcategorisation classes (SCCs) – depending on whether we take into account the linear order of the arguments in the sentence (SCF) or not (SCC) [18]. SCFs and SCCs respond to different needs: since they are very detailed, SCFs prove to be useful in studying Latin linear order, whereas SCCs present the traditional notion of verbal valency in terms of number and type of verbal arguments, and are more directly comparable with subcategorisation frames as they are described in the literature. Both SCFs and SCCs indicate the functional label for each argument, along with its morphological features. [19] In addition to such syntactic and morphological information, the lexical fillers of the arguments are recorded as lemmas; this can be used while acquiring the verb's selectional preferences.

17. As described in what follows, the expression "subcategorisation structures" is meant to have a more general sense than "subcategorisation frames"; this explains the different names.

18. The order of the arguments in SCC is alphabetical.

19. The shown morphological features are case for nouns and adjectives, and mood for verbs.

Table 7 contains the SCF and the SCC for the occurrence of the verb *do* (referred to as "V" in SCF) in [1], along with its voice. [20] The dependency tree of [1] is represented in Figure 3. In this case, the active form of the verb *do* is preceded by a nominative

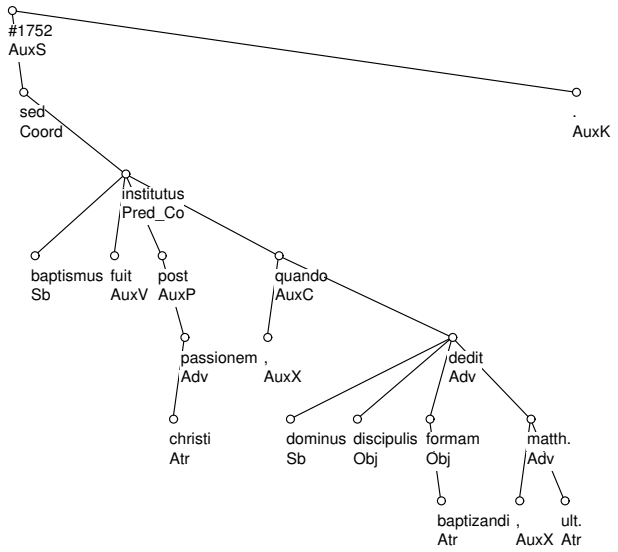

Figure 3. *Dependency tree of [1]*

verb	voice	subcat structures	morph. feat.	lexical fillers
do	A	SCF:Sb+Obj+Obj+V	Sb:nom	Sb:dominus
			Obj:dat	Obj:discipulus
		SCC:Obj,Obj,Sb	Obj:acc	Obj:forma

Table 7. *The SCF and the SCC for the verb* do *in [1]*

subject (*dominus*) and followed by a dative object (*discipulis*) and an accusative object (*formam*).

In order to account for the intermediate nodes that may intervene between the verbal headword and the argument nodes – prepositions (AuxP), conjunctions (AuxC) and coordinating (Coord) or apposing elements (Apos) –, we decided to record this information into special subcategorisation structures, namely SCF_1 and SCC_1, SCF_2, and SCC_2. After indexing the coordinating or apposing elements, SCF_1 and SCC_1 record the path from the verbal head to the argument nodes. In a similar vein, SCF_2

20. In table 7 "A" stands for "active".

and SCC_2 assign the same indices to those argument nodes depending on shared coordinating or apposing nodes, without recording the path along the tree. These indices have been adopted in order to disambiguate subcategorisation structures where more coordinated or apposed objects (Obj_Co or Obj_Ap) can refer to different verbal arguments. For instance, in sentence 2, [21] the pronoun *illa* refers to the noun *ars*, previously mentioned in the text. The verbal node *reservat* is coordinated with *reservant* through the coordinating node *et*. Both verbal nodes are annotated with the label Adv, since they are the predicates of subordinate adverbial clauses.

et	*illa*	*reservat*	*ulterius*	
and	that-NOM.F.SG	reserve-PRS.3SG	further-ADV.COMP	
"and that [art] reserves the final				
finem,	*scilicet*	*usum*	*navis,*	
end-ACC.F.SG,	namely-ADV	use-ACC.M.SG	ship-GEN.F.SG	[2]
end, i. e. the use of the ship,				
arti	*superiori,*	*scilicet*	*gubernatoriae*	
art-DAT.F.SG	superior-DAT.F.SG	namely-ADV	of governing-DAT.F.SG	
to the superior art, i.e. that of governing"				

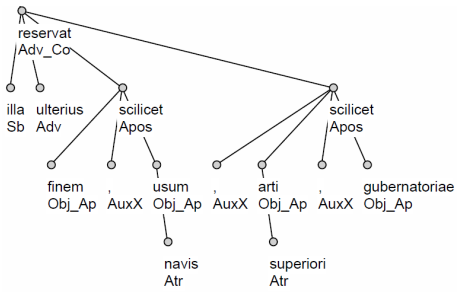

Figure 4. *Dependency tree of [2]*

Table 8 gives the subcategorisation structures of the verbal lemma *reservo* in [2], where the instance of *reservo* is triargumental, as shown by SCC_3. This requires disambiguating the two pairs of objects that receive the same functional label Obj_Ap (*finem*, *usum*, and *arti*, *gubernatoriae*), and is done by assigning them different in-

21. Thomas, *Super Sententiis Petri Lombardi*, IV, Distinctio 7, Quaestio 3, Articulus 1B, Solutio, 7-4.8-7.

SCF_1	Sb+V+(Apos[1])Obj_Ap+(Apos[1])Obj_Ap+(Apos[2])Obj_Ap+ +(Apos[2])Obj_Ap
SCC_1	(Apos[2])Obj_Ap,(Apos[2])Obj_Ap+(Apos[1])Obj_Ap,(Apos[1])Obj_Ap,Sb
SCF_2	Sb+V+Obj_Ap[1]+Obj_Ap[1]+Obj_Ap[2]+Obj_Ap[2]
SCC_2	Obj_Ap[1],Obj_Ap[1],Obj_Ap[2],Obj_Ap[2],Sb
SCC_3	Obj,Obj,Sb

Table 8. *Subcategorisation structures of* reservo *in [2]*

dices. [22] The apposing element *scilicet*, heading the two direct objects *finem* and *usum*, is assigned index 1, whereas the second *scilicet*, heading the two indirect objects *arti* and *gubernatoriae*, is assigned index 2 (see SCF_1, SCC_1, SCF_2 and SCC_2 in table 8). In addition, SCF_1 and SCC_1 record the full paths from the verbal head V to its argument nodes Obj_Ap.

The different available subcategorisation structures (SCFs and SCCs) can be used according to research interests. For example, if we want to study the coordinated objects in our corpus and their order, we may be interested in SCF_1 structures and their lexical fillers. Conversely, if we want to study the actual arguments of the verbs, disregarding both their linear order in the sentences and the path from the verbal headword along the tree, we may focus on SCC_3 structures, thus associating each verb type with its SCC_3s. The voices of the verb's forms and the morphological features of its arguments (represented in square brackets) can also be accounted for, along with the corresponding (absolute and relative) frequencies, as shown in table 9. [23]

Table 9 shows that two-thirds (35) of the 52 occurrences of *do* are in the active form and associated with at least one of the following arguments: Obj[abl], Obj[dat] and Sb[nom]. This reflects the intuition that this verb has three possible argument slots to be filled: a nominative subject, an accusative direct object and a dative indirect object. [24] In regards to the passive usages, [25]the objects in ablative (Obj[abl]) always depend on the preposition *a/ab* ("by") in our data. They represent agentive arguments, corresponding to the subjects in the active form.

4.3.3. *Quantitative data*

The IT-TB currently contains 8,060 verbal tokens and 432 verbal lemmas, following a Zipfian distribution where frequencies range from 1,876 (*sum*, "to be") to 1 (158

22. If we provide SCC_3 with morphological features, we notice that the two pairs of objects have also different cases: SCC_3=Obj[accusative],Obj[dative],Sb[nominative].

23. The abbreviations in table 9 stand for: "nominative" ("nom"), "genitive" ("gen"), "dative" ("dat"), "accusative" ("acc"), "ablative" ("abl") and "infinitive" ("inf").

24. This is confirmed by (Happ, 1976, p. 559), where two possible subcategorisation frames for *do* are reported: the first requires two obligatory arguments (nominative and accusative) and the second adds a third dative argument.

25. Referred to as "P" in table 9.

voice	SCC_3	fr.(*do*,SCC_3)	rel.fr.(*do*,SCC_3)
A	Obj[acc],Sb[nom]	13	0.25
A	Obj[acc],Obj[dat],Sb[nom]	9	0.17
A	Obj[acc],Obj[dat]	7	0.13
A	Obj[acc]	6	0.12
P	Sb[nom]	5	0.10
P	V	2	0.04
P	Obj[abl]	2	0.04
P	Obj[dat],Sb[subj]	2	0.04
P	Obj[dat]	1	0.02
P	Obj[abl],Sb[nom]	1	0.02
P	Obj[abl],Obj[dat],Sb[nom]	1	0.02
P	PNom[inf],Sb[nom]	1	0.02
P	Obj[dat],Sb[nom]	1	0.02
P	Obj[dat],Sb[inf]	1	0.02

Table 9. *Frequency counts of SCC_3 structures for the verb* do

verbs) with a high standard deviation (94.6); this shows a large variability in the data, the average and the median of the frequencies being respectively 13.8 and 2.5.

The database queries of the valency lexicon search for every verbal occurrence in the treebank, collect its arguments, and represent them into one of the previously defined subcategorisation structures.[26] Table 10 shows the frequency counts of each subcategorisation structure in the treebank. From this table we can see that the SCF/SCC rate is 1.67 for SCF_1/SCC_1 and 2.18 for SCF_2/SCC_2. This shows the different granularity of subcategorisation frames and subcategorisation classes, and therefore their different frequency distribution.

subcategorisation structure	SCF_1	SCC_1	SCF_2	SCC_2	SCC_3
frequency	271	162	177	81	16

Table 10. *Frequency counts for different subcategorisation structures in IT-TB*

Following (Hinrichs and Telljohann, 2009), we decided to study if the number of subcategorisation structures per verb is correlated with the frequency of the verb. To test for such a correlation, we used the Spearman rank correlation test computed in R (R Development Core Team, 2008). This test converts the frequencies to ranks and is

26. For those cases where a verb does not exhibit any explicit argument in the treebank, the verbal occurrence is reported as "absolute". These absolute usages amount to 2094, which correspond to 26% of the total number of verb tokens.

subcategorisation structure	r_s	S	p-value
SCF_1	0.94	1315278	p-value<0.01
SCC_1	0.93	1735097	p-value<0.01
SCF_2	0.94	1330671	p-value<0.01
SCC_2	0.93	1775455	p-value<0.01
SCC_3	0.91	2144382	p-value<0.01

Table 11. *Correlation coefficient r_s, test statistic S (the sum of rank differences) and p-value for each test*

an ordinal version of the Pearson correlation test. Unlike the latter, the Spearman test is very robust against outliers and non-linear correlations.

For each subcategorisation structure we performed a Spearman rank test, with the results provided in table 11. All the p-values, which represent the likelihood of the rank pairs arising by chance, are significant at the 0.01 level, even after a Bonferroni correction. The r_s coefficients indicate a strong positive correlation for all the structures. In particular, the structures that take the sentence order into account (SCF_1 and SCF_2) display a higher correlation ($r_s = 0.94$) than the ones that do not (SCC_1, SCC_2 and SCC_3). If we look at the SCCs, we find that the first two have the same degree of correlation, whereas the last one performs worse, which is expected since it is more coarse-grained and contains the least information. Analysing the figures showing the verb frequencies plotted against the number of subcategorisation structures per verb (which are not reported here for reasons of space), we noticed that the correlation is higher for the low-frequency verbs. Since the corpus is finite, our results are indeed partially influenced by low-frequency verbs, which occur with very few frames. For example, if a verb is only seen once in the corpus, the number of its frames is 1; however, this does not mean that the verb could not appear with other frames in a larger corpus. This could partially explain why the correlation we found between the verb frequency and the frame count is so high. [27]

5. Conclusions and future work

In this paper we presented an overview of the IT-TB project. In particular, we described parsing procedures on IT-TB data and the IT-TB valency lexicon. The main task of the project now is to increase the amount of annotated data. This will be done semi-automatically, using the available trained parsers; further, having more data

27. To see how hapaxes and low-frequency verbs affected our results, we performed the same analysis on the most frequent verbs only and found a lower but still medium-sized correlation (in the 0.5-0.7 range). These results disagree with the ones described in (Hinrichs and Telljohann, 2009) for a much larger corpus (36,000 sentences), where the authors take into account the most frequent verbs only and find a weak correlation.

available can improve the performance of data-driven parsers and increase the size of the valency lexicon.

As far as parsing procedures are concerned, in the near future we foresee defining and applying features specific to Latin syntax. These features will be added by exploiting lexical information on the predicate-argument structure recorded in the valency lexicon. Better accuracy rates may also be obtained through methods of parsing combination.

As far as the IT-TB valency lexicon is concerned, we plan to make the lexicon available on-line through a graphical interface that can be integrated into the annotation tool. This way, the consistency of the annotation process can be tested and enforced, thanks to the information stored in the lexicon. The lexicon will also be enriched with valency information for nouns and adjectives.

In order to test the accuracy of the lexicon, it can be evaluated against other existing resources for Latin, such as Happ's list and traditional dictionaries and thesauri. A comparison with the Perseus "dynamic lexicon" may be also very interesting for contrastive and diachronic studies on Classical and Medieval Latin.

Following a broad approach in valency lexicons, a close connection between valency frames and word senses will be maintained in the development of lexicon entries: this means that each headword entry of our lexicon will consist of one or more SCFs and SCCs, one for each sense of the word.

6. Acknowledgements

We wish to thank Gard Jenset, Erik Norvelle and Daniela Viviani for their help.

7. References

Aquinas T., *Sancti Thomae Aquinatis, doctoris angelici, Ordinis praedicatorum Commentum in quatuor libros Sententiarum magistri Petri Lombardi, adjectis brevibus adnotationibus*, Fiaccadori, Parma, 1856-1858.

Attardi G., "Experiments with a Multilanguage Non-Projective Dependency Parser", *Proceedings of the Tenth Conference on Computational Natural Language Learning (CoNLL-X)*, Association for Computational Linguistics, New York City, p. 166-170, June, 2006.

Attardi G., Ciaramita M., "Tree Revision Learning for Dependency Parsing", *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, Association for Computational Linguistics, Rochester, New York, p. 388-395, April, 2007.

Attardi G., Dell'Orletta F., Simi M., Chanev A., Ciaramita M., "Multilingual Dependency Parsing and Domain Adaptation using DeSR", *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, Association for Computational Linguistics, Prague, Czech Republic, p. 1112-1118, June, 2007.

Bamman D., Crane G., "The Latin Dependency Treebank in a cultural heritage digital library", *Proceedings of the Workshop on Language Technology for Cultural Heritage Data (LaTeCH 2007)*, Prague, Czech Republic, p. 33-40, 2007.

Bamman D., Crane G., "Building a Dynamic Lexicon from a Digital Library", *Proceedings of the Eighth ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2008)*, Pittsburgh, USA, 2008.

Bamman D., Crane G., Passarotti M., Raynaud S., "A Collaborative Model of Treebank Development", *Proceedings of the Sixth Workshop on Treebanks and Linguistic Theories (TLT 2007)*, Bergen, Norway, p. 1-6, 2007a.

Bamman D., Crane G., Passarotti M., Raynaud S., Guidelines for the Syntactic Annotation of Latin Treebanks, Technical report, Tufts Digital Library, 2007b.

Bamman D., Passarotti M., Busa R., Crane G., "The annotation guidelines of the Latin Dependency Treebank and Index Thomisticus Treebank. The treatment of some specific syntactic constructions in Latin", *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC 2008)*, Marrakech, Morocco, 2008a.

Bamman D., Passarotti M., Crane G., "A Case Study in Treebank Collaboration and Comparison: Accusativus cum Infinitivo and Subordination in Latin", *Prague Bulletin of Mathematical Linguistics*, vol. 90, p. 109-122, 2008b.

Böhmová A., Hajič J., Hajičová E., Hladká B., "The Prague Dependency Treebank: A Three-Level Annotation Scenario", *in* A. Abeillé (ed.), *Treebanks: building and using parsed corpora*, Kluwer Academic Publishers, Dordrecht, NL, p. 103-128, 2003.

Buchholz S., Marsi E., "CoNLL-X Shared Task on Multilingual Dependency Parsing", *CoNLLX*, SIGNLL, 2006.

Bühler K., *Sprachtheorie: die Darstellungs-funktion der Sprache*, Jena: Gustav Fischer, Stuttgart, 1934.

Busa R., *Index Thomisticus: sancti Thomae Aquinatis operum omnium indices et concordantiae, in quibus verborum omnium et singulorum formae et lemmata cum suis frequentiis et contextibus variis modis referuntur quaeque / consociata plurium opera atque electronico IBM automato usus digessit Robertus Busa SJ.*, Frommann - Holzboog, Stuttgart – Bad Cannstatt, 1974-1980.

Carroll J., Briscoe T., Sanfilippo A., "Parser Evaluation: a Survey and a New Proposal", *Proceedings of the First International Conference on Language Resources and Evaluation (LREC 1998)*, Granada, Spain, p. 447-454, 1998a.

Carroll J., Minnen G., Briscoe T., "Can subcategorization probabilities help a statistical parser?", *Proceedings of the Sixth ACL/SIGCDAT Workshop on Very Large Corpora*, Montreal, Canada, 1998b.

Covington M. A., A dependency parser for variable-word-order languages, Technical report, Artificial Intelligence Programs, University of Georgia, 1990.

Covington M. A., "A Fundamental Algorithm for Dependency Parsing", *39th Annual ACM Southeast Conference*, Athens, Georgia, USA, p. 95-102, 2001.

Crane G., "Generating and Parsing Classical Greek", *Literary and Linguistic Computing*, vol. 6, n° 4, p. 243-245, 1991.

Crane G., Chavez R. F., Mahoney A., Milbank T. L., Rydberg-Cox J. A., Smith D. A., Wulfman C. E., "Drudgery and deep thought: Designing a digital library for the humanities", *Communications of the ACM*, vol. 44, n° 5, p. 34-40, 2001.

Denooz J., "La banque de données du laboratoire d'analyse statistique des langues anciennes (LASLA)", *Le Médiéviste et l'ordinateur*, vol. 33, p. 14-20, 1996.

Hajič J., Panevova J., Uresova Z., Bemova A., Annotations at analytical level: Instructions for annotators (English translation by Z. Kirschner), Technical report, UFAL MFF UK, Prague, 1999.

Hajič J., Panevová J., Urešová Z., Bémová A., Kolárová-Reznícková V., Pajas P., "PDT-VALLEX: Creating a Large Coverage Valency Lexicon for Treebank Annotation", *in* J. Nivre, E. Hinrichs (eds), *Proceedings of the Second Workshop on Treebanks and Linguistic Theories (TLT 2003)*, vol. 9, Växjö University Press, Växjö, Sweden, p. 57-68, 2003.

Happ H., *Grundfragen einer Dependenz-Grammatik des Lateinischen*, Vandenhoeck & Ruprecht, Goettingen, 1976.

Haug D. T. T., Jøndal M. L., "Creating a Parallel Treebank of the Old Indo-European Bible Translations", *Proceedings of Language Technologies for Cultural Heritage Workshop - LREC 2008*, Marrakech, Morocco, p. 27-34, 2008.

Hinrichs E. W., Telljohann H., "Constructing a Valence Lexicon for a Treebank of German", *Proceedings of the Seventh International Workshop on Treebanks and Linguistic Theories (TLT 7)*, Groningen, The Netherlands, p. 41-52, 2009.

Koch U., The Enhancement of a Dependency Parser for Latin, Technical Report n° AI-1993-03, Artificial Intelligence Programs, University of Georgia, 1993.

Korhonen A., Krymolowski Y., Briscoe T., "A Large Subcategorization Lexicon for Natural Language Processing Applications", *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy, 2006.

Koster C. H. A., "Affix Grammars for natural languages", *Attribute Grammars, Applications and Systems, International Summer School SAGA*, vol. 545 of *Lecture Notes in Computer Science*, Prague, Czech Republic, 1991.

Koster C. H. A., "Constructing a Parser for Latin", *Computational Linguistics and Intelligent Text Processing*, Lecture Notes in Computer Science, Springer, Berlin - Heidelberg, p. 48-59, 2005.

Kromann M. T., "The Danish Dependency Treebank and the underlying linguistic theory", *in* J. Nivre, E. Hinrichs (eds), *Proceedings of the Second Workshop on Treebanks and Linguistic Theories (TLT 2003)*, Växjö, Sweden, 2003.

Lin D., "A dependency-based method for evaluating broadcoverage parsers", *Proceedings of the IJCAI-95*, Montreal, Canada, p. 1420-1425, 1995.

McDonald R., Pereira F., "Online Learning of Approximate Dependency Parsing Algorithms", *In Proc. of EACL*, p. 81-88, 2006.

McDonald R., Pereira F., Ribarov K., Hajič J., "Non-projective dependency parsing using spanning tree algorithms", *HLT '05: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, Association for Computational Linguistics, Morristown, NJ, USA, p. 523-530, 2005.

McGillivray B., Passarotti M., "The Development of the Index Thomisticus Treebank Valency Lexicon", *Proceedings of the Workshop on Language Technology and Resources for Cultural Heritage, Social Sciences, Humanities, and Education (LaTeCH-SHELT&R 2009)*, Athens, Greece, 2009.

Messiant C., Korhonen A., Poibeau T., "LexSchem: A Large Subcategorization Lexicon for French Verbs", *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC 2008)*, Marrakech, Morocco, 2008.

Mírovský J., "Netgraph: a Tool for Searching in Prague Dependency Treebank 2.0", *in* J. Hajič, J. Nivre (eds), *Proceedings of the Fifth Workshop on Treebanks and Linguistic Theories*, Prague, Czech Republic, p. 211-222, 2006.

Nilsson J., Nivre J., "MaltEval: an Evaluation and Visualization Tool for Dependency Parsing", *in* E. L. R. A. (ELRA) (ed.), *Proceedings of the Sixth International Language Resources and Evaluation (LREC 2008)*, Marrakech, Morocco, May, 2008.

Nivre J., "An Efficient Algorithm for Projective Dependency Parsing", *Proceedings of the Eighth International Workshop on Parsing Technologies (IWPT)*, p. 149-160, 2003.

Nivre J., "Incrementality in Deterministic Dependency Parsing", *in* F. Keller, S. Clark, M. Crocker, M. Steedman (eds), *Proceedings of the ACL Workshop Incremental Parsing: Bringing Engineering and Cognition Together*, Association for Computational Linguistics, Barcelona, Spain, p. 50-57, July, 2004.

Nivre J., Nilsson J., "Pseudo-projective dependency parsing", *ACL '05: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, Association for Computational Linguistics, Morristown, NJ, USA, p. 99-106, 2005.

Nivre J., Scholz M., "Deterministic dependency parsing of English text", *COLING '04: Proceedings of the 20th International Conference on Computational Linguistics*, Association for Computational Linguistics, Morristown, NJ, USA, p. 64, 2004.

Passarotti M., "LEMLAT. Uno strumento per la lemmatizzazione morfologica automatica del latino", *in* F. Citti, T. Del Vecchio (eds), *From Manuscript to Digital Text. Problems of Interpretation and Markup. Proceedings of the Colloquium (Bologna, June 12th 2003)*, Rome, Italy, p. 107-128, 2007a.

Passarotti M., "Verso il Lessico Tomistico Biculturale. La treebank dell'Index Thomisticus", *in* R. Petrilli, D. Femia (eds), *Il filo del discorso. Intrecci testuali, articolazioni linguistiche, composizioni logiche. Atti del XIII Congresso Nazionale della Società di Filosofia del Linguaggio*, Viterbo, Italy, p. 187-205, 2007b.

Pinkster H., *Latin Syntax and Semantics*, Routledge, London, UK, 1990.

R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria. 2008, ISBN 3-900051-07-0.

Schinke R., Greengrass M., Robertson A., Willett P., "Retrieval of Morphological Variants in Searches of Latin Text Databases", *Computers and the Humanities*, vol. 31, p. 409-432, 1998.

Tesnière L., *Éléments de syntaxe structurale*, Editions Klincksieck, Paris, France, 1959.

Titov I., Henderson J., "A Latent Variable Model for Generative Dependency Parsing", *Proceedings of the Tenth International Conference on Parsing Technologies*, Association for Computational Linguistics, Prague, Czech Republic, p. 144-155, June, 2007.

Urešová Z., *The Verbal Valency in the Prague Dependency Treebank from the Annotator's Point of View*, Jazykovedný ústav L. Štúra, SAV, Bratislava, Slovakia, 2004.

Van der Beek L., Bouma G., Malouf R., van Noord G., "The Alpino Dependency Treebank", *in* M. Theune, A. Nijholt, H. Hondorp (eds), *Proceedings of the Twelfth Meeting of Computational Linguistics in the Netherlands (CLIN 2001)*, Rodopi, Amsterdam, p. 8-22, 2002.