POS tagset design for Italian

Raffaella Bernardi[*], **Andrea Bolognesi**[◇], **Corrado Seidenari**[◇], **Fabio Tamburini**[◇]

[*]KRDB, Free University of Bolzano Bozen, Italy
bernardi@inf.unibz.it

[◇]CILTA, University of Bologna, Italy
{a.bolognesi, c.seidenari, f.tamburini}@cilta.unibo.it

Abstract

We aim to automatically induce a PoS tagset for Italian by analysing the distributional behaviour of Italian words. To this end, we propose an algorithm that (a) extracts information from loosely labelled dependency structures that encode only basic and broadly accepted syntactic relations, namely Head/Dependent and the distinction of dependents into Argument vs. Adjunct, and (b) derives a possible set of word classes. The paper reports on some preliminary experiments carried out using the induced tagset in conjunction with state-of-the-art PoS taggers. The method proposed to design a proper tagset exploits little, if any, language-specific knowledge: hence it is in principle applicable to any language.

1. Introduction

The work presented in this paper is part of a project aiming to annotate CORIS/CODIS (Rossini Favretti et al., 2002), a 100-million-word synchronic corpus of contemporary written Italian, with part-of-speech (PoS) tags.

Italian is one of the languages for which a set of annotation guidelines has been developed in the context of the EAGLES project (Monachini, 1995). Several research groups have worked on PoS annotation to develop treebanks, such as VIT (Venice Italian Treebank (Delmonte, 2004)) and TUT (Turin University Treebank) (Bosco et al., 2000; Bosco, 2003) and morphological analysers such as of XEROX. A comparison of the tag sets used by these groups with Monachini's guidelines reveals that though there is general agreement on the main parts of speech to be used, considerable divergence exists when it comes to the actual classification of Italian words with respect to these main PoS classes.

The main categories identified within the EAGLES project are nouns, verbs, adjectives, adverbs, determiners, pronouns, articles, adposition, conjunctions, numerals, interjections and residuals. The actual tagset is then obtained by further subdividing these categories by means of semantics or morpho-syntactic criteria. The tagsets used by XEROX, VIT, and TUT could be said to quite strictly respect the main categorization.

The proposed tagsets differ, however, on the criteria used for subdividing the main classes and hence are rather different. For instance, VIT is much more fine grained in the distinction of nouns: instead of the classical distinction used by Monachini into proper and common nouns, this class is divided into subclasses semantically motivated, viz. colour (NC), factive (NF), temporal (NT), human (NH). Furthermore, both VIT and XEROX have word specific tags, e.g. VIT tags the prepositions 'di' and 'da' as 'PD' and 'PDA', respectively; similarly, XEROX uses 'CONNCHE' to tag the word 'che' both when used as relative pronoun and as conjunction. Finally, the homogeneity of the main classes distinction does not correspond to an equivalent homogeneity at the level of word assignments to these classes.

The classes for which differences of opinion are most evident are adjectives, determiners and adverbs. For instance, words like 'molti' (many) have been classified as indefinite determiners by Monachini, as plural quantifiers by XEROX and indefinite adjectives by VIT and TUT. These differences will then influence the kind of conclusions one can draw from the annotated corpus since they do not boil down to simply terminological differences resolvable by a mere one-to-one relabelling or by mapping different classes into a greater one. Another illustrative case is that of a very frequent word like 'stesso' (same). In VIT 'stesso', in its adjectival usage, is grouped with words like 'quello' (that) or 'questo' (this) within demonstrative adjectives (DIM). In XEROX, while 'quello' and 'questo' are tagged as determiners (DETSG, DETPL), 'stesso' is tagged as adjective (ADJSG, ADJPL) together with the large lexical class of the qualifying adjectives.

These factors drove us to propose to semi-automatically induce the word classes. Section 2. outlines briefly the methods and algorithms used to induce a PoS tagset for Italian: for space reason we heavily refer to an early work (Bernardi et al., 2005) for the detailed description of the induction algorithm. Section 3. shows some preliminary results we have obtained on tagset definition. Section 4. outlines the results obtained using the proposed tagset with conventional tagging techniques, while section 5. draws some provisional conclusions.

2. The proposed PoS induction method

Our aim is to automatically derive an empirically founded PoS classification making no *a priori* assumptions about the PoS classes to be distinguished.

Early approaches to this problem were based on the hypothesis that if two words are syntactically and semantically different, they will appear in different contexts. There are a number of studies based on this hypothesis in the fields of both computational linguistics and cognitive science aiming at building automatic or semi-automatic procedures for clustering words (Brill and Marcus, 1992; Pereira et al., 1993; Schütze, 1993; Clark, 2000; Redington et al., 1998;

Gobet and Pine, 1997). These works examine the distributional behaviour of target words by comparing the lexical distribution of their respective collocates and by using quantitative measures of distributional similarity.

The main drawback of these techniques is the limited context of analysis. Information is collected from a restricted context of, for instance, ± 3 words which can conceal syntactic dependencies longer than the context interval.

Our approach to solving this problem is to use basic syntactic relations together with distributional information. A basic distinction of word classes is induced by means of Brill's algorithm (Brill and Marcus, 1992) (described in (Tamburini et al., 2002)). Three main uncontroversial classes emerge from this broad range process: nouns (N), verbs (V) and all the others (X). This is an empirical statement of a widely accepted distinction in linguistic studies.

This classification is further refined by means of minimal syntactic information. We extract this information from loosely labelled dependency structures that encode only basic and broadly accepted syntactic relations, namely Head/Dependent, and the distinction of dependents into Argument/Adjunct. A large number of specific syntactic descriptions per word are exploited to identify differences in the syntactic behaviour of words. In associating lexical items with rich descriptions, our approach is, to some extent, related to supertags (Bangalore and Joshi, 1999).

Our dependency structures are derived from TUT (Bosco et al., 2000; Bosco, 2003). The treebank currently includes about 1500 sentences organized in different sub-corpora from which we converted the dependency trees maintaining only the basic syntactic information required for this study. Words are marked as N (nouns), V (verbs) or X (all others) according to the results obtained in (Tamburini et al., 2002). We use $<$ $>$ to mark Head-Argument relation and $\ll$ and $\gg$ to mark Head-Adjunct relation where the arrows point to the Head. From these dependency structures we extract syntactic type assignments by projecting dependency links onto formulas. Formulas are built out of $\{<, >, \ll, \gg, N, X, V, Lex\}$ where the symbol Lex stands for the word the formula has been assigned to. The formal description of the type resolution algorithm, that assigns a syntactic type to every word in sentence, has been slightly modified with respect to the method presented in (Bernardi et al., 2005).

Type Resolution Let $W = \langle w_1, ..., w_n \rangle$ stand for an ordered sequence of words in a given sentence and let $w_j = \langle orth_j, bl_j, t_j \rangle$ stand for a word in the sentence, where $orth_j$, $bl_j \in \{N, V, X\}$ and t_j represent the orthographic transcription, the basic label and the type of the j-th word respectively. Let $E = \{\langle R, w_i, w_k \rangle\}$ be the set of edges where $R \in \{<, >, \ll, \gg\}$ is ordered by $|k - i|$ in ascending order. Given a dependency structure represented by means of W and E,

$- \forall w_j \in W, \ t_j = Lex$
$-$ foreach $\langle R, w_i, w_j \rangle \in E$

$$
\begin{array}{lll}
\text{if } R =' >' & \langle w_j, bl_j, t_j \rangle \rightsquigarrow \langle w_j, bl_j, bl_i > t_j \rangle & (\dagger) \\
& \langle w_i, bl_i, t_i \rangle \rightsquigarrow \langle w_i, bl_i, t_i >^* bl_j \rangle & (\diamond) \\
\text{if } R =' <' & \langle w_i, bl_i, t_i \rangle \rightsquigarrow \langle w_i, bl_i, t_i < bl_j \rangle & (\dagger) \\
& \langle w_j, bl_j, t_j \rangle \rightsquigarrow \langle w_j, bl_j, bl_i <^* t_j \rangle & (\diamond) \\
\text{if } R =' \ll' & \langle w_j, bl_j, t_j \rangle \rightsquigarrow \langle w_j, bl_j, bl_i \ll t_j \rangle & (\dagger) \\
& \langle w_i, bl_i, t_i \rangle \rightsquigarrow \langle w_i, bl_i, t_i \ll^* bl_j \rangle & (\diamond) \\
\text{if } R =' \gg' & \langle w_i, bl_i, t_i \rangle \rightsquigarrow \langle w_i, bl_i, t_i \gg bl_j \rangle & (\dagger) \\
& \langle w_j, bl_j, t_j \rangle \rightsquigarrow \langle w_j, bl_j, bl_i \gg^* t_j \rangle & (\diamond)
\end{array}
$$

where the operator $\rightsquigarrow$ replaces the first item with the second in W. Each rule above is composed by two $\rightsquigarrow$ operations: if we apply the ($\dagger$) ones we will obtain the 'nuclear types' only, while if we apply both ($\dagger$) and ($\diamond$) rules we will obtain what we call 'extended types'. Figure 1 shows a type resolution example for two simple dependency graphs outlining both nuclear and extended types.

The type resolution procedure, creating a set of word-type pairs, transforms the dependency treebank into a lexicon in which every word contained in the treebank exhibit all the syntactic types emerged from the type resolution process.

Initial dep. structure	Final type resolution	
	il:	$Lex < N$
		$(-)$
	libro:	Lex
		$(X <^* Lex \ll^* X)$
il libro rosso X N X (the) (book) (red)	rosso:	$N \ll Lex$
		$(-)$
	Carlo:	Lex
		$(Lex >^* X)$
	e:	$N > Lex < N$
		$(N > Lex < N >^* V)$
Carlo e Carla corrono N X N V (Carlo) (and) (Carla) (run)	Carla:	Lex
		$(X <^* Lex)$
	corrono:	$X > Lex$
		$(-)$

Figure 1: Type resolution examples. Nuclear types and extended types (in parenthesis).

The algorithm proposed creates pairs of words and syntactic types, by means of the type resolution outlined above (phase I) and connects each pair in accordance with syntactic similarities between them, producing an extensive *inclusion graph* as showed in figure 2 (phase II). The algorithm then exploits statistical information extracted from the inclusion graph, namely word and type frequencies, in order to prune it and to extract a complete set of PoS-class hypotheses (phase III).

Figure 3 shows a flow chart which summarizes the three phases of our algorithm.

A detailed mathematical description of the algorithm phase II and III can be found in (Bernardi et al., 2005).

The final output of the three phase system is expected to help the linguist to define a proper tagset to apply during the annotation phase as well as when searching the annotated corpus. The resulting PoS classification can be organized as a hierarchy with inclusion relations, as we can see in the following sections, thus a more powerful search interface can be constructed to help the user extract the relevant information from the annotated corpus.
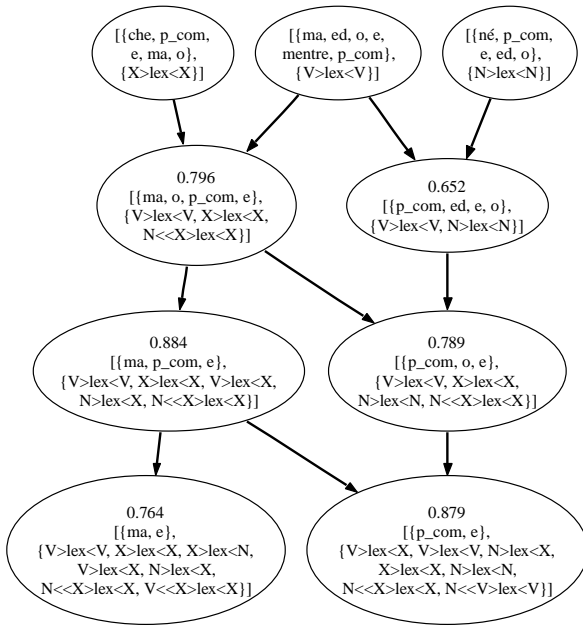
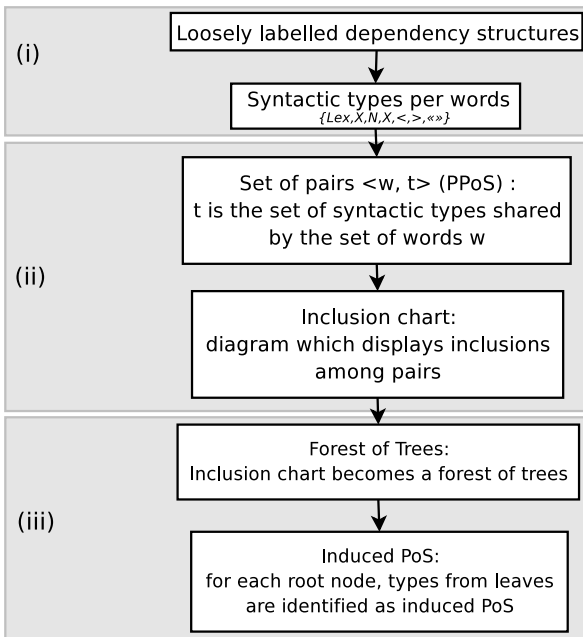Figure 2: An example of *inclusion graph*.

Figure 3: Algorithm Architecture.

3. PoS Tagset induction

The tagset induction procedure has been configured as a multistep process. The algorithm described in the previous section has been applied to the lexicon derived from the entire dependency treebank with the aim of further subdivide the X class in more categories. After applying the induction algoritm the first time, five categories emerged quite clearly, added to our original classes, nouns (N) and verbs (V): adjectivals (ADJ), adverbials (ADV), a category of entities, relatives (RELS), subordinators head of a modifying clause (SUBORD_M), coordinators (COORD) and again a group of other classes which are difficult to interpret. This subdivision has been obtained by exploiting the nuclear types only.

In the second step we removed the word-type pair belonging to the five categories outlined before from the lexicon produced by the type resolution process and re-applied phase II and III of the type induction process on the remaining word-type pairs, using the extended types instead of the nuclear types as in the first step.

Table 1 shows the results obtained after this two-step process.

3.1. Analysis of the induced PoS

At the moment we have arrived at the point of inducing a first level distinction of PoS tags comparable with the twelve EAGLES main classes listed above. Further studies will be carried out to subdivide them into more fine grained distinctions. Four tags correspond straightforwardly to the ones in the EAGLES project, namely nouns, verbs, adjectives, adverbs, differences instead arise with respect to the other tags as described below.

ENTITIES: This is a class of pronouns or expressions that behave like pronouns. For instance, 'coloro' (those) in the following example[1]

- . . .tutti **coloro** che offrono aiuto. . .

RELS: This contains both a) relative pronouns and b) adverbs when behaving syntactically in the same way.

a) . . .ai terreni su **cui** esistevano. . .

b) . . .vicino all'università **dove** nel '90 scoppiò la rivolta. . .

SUBORD: This contains expressions that bridge a main phrase and a subordinate. Two different classes emerged from the type induction process: a) the subordinators which are head of a clause modifying the main one (SUBORD_M, e.g. 'quando' (when), 'perché' (why)) and b) the subordinators which are head of a clause which is dependent on a verbal head (SUBORD_A, e.g 'che' (that), 'di' (to)) as illustrated by the following examples:

a) . . .si applicano anche **quando** si tratta di togliere un ingombro. . .

b) . . .salvo che esigenze tecniche impongano **di** costruirlo. . .

COORD: This is a class that includes the classical coordinators, e.g. 'e' (and), 'o' (or), 'ma' (but).

PREP_M: This contains prepositions, for instance 'attraverso', 'secondo', 'con', 'sul', 'nel', 'di', 'degli', which in construction mainly with nouns, modify a) verbs or b) nouns as exemplified below. These expressions have been found in different constructions too, hence they have also received another tag as explained in the ARG class.

a) . . .provvedere **in** tempo. . .

b) . . .proporzione **del** vantaggio. . .

---

[1]Our algorithm assigned to this class also predicative components of copulative structures. They received the same syntactic description of e.g. pronouns as they are in the same dependency relation with the verbal head. Most probably this problem would have not arisen if we considered a finer grained distinction among verbs.

Proposed PoS Label		Induced type cluster	Prototypical words
Nouns		N	nuvola, finestra, tv
Verbs		V	stupire, raggiunto, concludendo, abbiamo, allora,
X	Adverbials	V≪Lex, Lex≫V, Lex≫X	appena, decisamente, ieri, mai,
			molto, persino, rapidamente, presto, troppo
	Adjectivals	Lex≫N, N≪Lex, X≪Lex	economici, elettorale, forti, giovane, grande,
			idrica, importanti, nuove, piccolo, positiva, suo,
			terzo, ufficiale, ultima, vicino
	Coordinators	V>Lex<V, N>Lex<N, X>Lex<X, N>Lex<X,	e, ed, ma, mentre, o, ovvero,
		X>Lex<N, V>Lex<X, X>Lex<V, V≪X>Lex<X,	oppure, sia
		X>Lex<X≫V, N≪X>Lex<X	
	Entities	Lex	ci, di_più, in_salvo, io, inferocito, noi, ti,
	Relatives	N>Lex	che, cui, dove, quale
	Subordinators_MOD	Lex<V≫X, V≪Lex<V	affinché, come, dopo, mentre, perché, qualora
	Subordinators_ARG	V<*Lex<V, X<*Lex<V, Lex<V>*X, V<*Lex<X,	a, che, da, di, per, se
		X<*Lex<X, N≪Lex<V, X≪Lex<V	
	ARG	Lex<N, Lex<X>*V, X<*Lex<N, X<*Lex<N,	ARG_Det: alcuni, gli, il, l', le, questa, qualche,
		V<*Lex<N, X<*Lex<V ...	quattro, un
			ARG_Prep: a, alla, con, da, della, di, nei, sul
	Prepositionals_POLI	V≪Lex<X, Lex<X≫V, N≪Lex<X, Lex<X≫X	contro, dopo, durante, nonostante, secondo, verso
	Prepositionals_NM	N≪Lex<N, X≪Lex<N, N<*Lex<N, N<*Lex<X	agli, degli, del, della, di, nei, nelle, sui, sulla
	Prepositionals_VM	V≪Lex<N, Lex<N≫V, V≪X≫*Lex<N	alla, all', con, dagli, nella, nell', sulle, tra

Table 1: The PoS classification emerged automatically from the two-step class induction process.

Our algorithm created three different classes: 1) PREP_POLI, 'attraverso', 'secondo', 'contro', etc. which take a noun phrase or a prepositional phrase and modify a verb[2]; 2) PREP_NM, 'del', 'degli' etc. which take a noun and modify a noun; 3) PREP_VM, 'nella', 'sul' which take a noun and modify a verb. The last two classes are exemplified by a) and b) above, the first one is illustrated by c) below:

c) …protestare **contro** il Governo…

ARG: This class includes all those expressions that are distributionally close to articles. They are the head of a phrase dependent mainly on a verb. Hence, members of this class are expressions like, 'il' (the) but also, 'mio' (my) and 'di' (of) when occurring, for example, in the construction shown in a) and b), respectively.

a) …l'unica volta che **mio** padre mi portò al cinema…

b) …si parla **di** 250-300 milioni di dollari…

As stated above, the classification of ARG partially resulted in a fusion between word classes such as determiners and prepositions which are traditionally thought of as being neatly divided. This is due to the fact that, in assigning words to word classes, we relied on basic syntactic information (see section 2.). As a result, the ARG class did not seem fine grained enough and, more crucially, not user oriented enough considering the purpose of tagging a corpus like CORIS/CODIS which is intended as a reference resource for Italian language. So we proposed to split the ARG class, following morphological criteria, into two subclasses: ARG_Det and ARG_Prep including mainly determiners and prepositions respectively.

Another point concerning human intervention on the automatically induced tagset deserves some further comments. Our algorithm originally produced two separate classes of adjectives depending on whether they are of predicative or attributive distribution with respect to the word they modify. Considering the sparseness of data, such a sharp distinction on a distributional basis seemed too premature in

this experimental phase of our project. So we decided to manually assemble the two classes into a single one.

Finally, for the tagging experiments we decided to test 2 different tagsets. Tagset 1 (TS1), made of 15 tags, comprises 3 different tags for modifying prepositionals (i. e. PREP_NM, PREP_VM and PREP_POLI) which were the original outcome of our algorithm. Tagset 2 (TS2), as described above, comprises a single superordinate class, PREP_M, for modifying prepositionals. TS2 therefore is made of just 13 tags. Both tagsets include ARG_Det and ARG_Prep as two separate tags.

3.2. Differences from the EAGLES proposal

As the reader might have noticed, substantial differences result when comparing these classes with the more widely accepted ones. First of all, traditional pronouns divided in the Eagles project into possessive, demonstratives, indefinite, interrogatives, exclamatives, and personal are clustered together mainly with expressions traditionally considered as numerals. Similarly, relative expressions, due to their specialized syntactic behavior have consequently been classified alone. This class groups relative pronouns together with similar expressions like 'dovunque' or 'dove', traditionally tagged as adverbs (occasionally specified as relative adverbs). Secondly, conjunctions, divided by Monachini into coordinators and subordinators, are clustered differently. On the one hand, the class of coordinators correspond to our COORD, but on the other hand the class of subordinators does not have a direct correspondence in our tagset: SUBORD_A and SUBORD_M also contain expressions like 'di', 'a' and 'per' traditionally considered prepositions. Adpositions (i.e. the prepositions), numerals and determiners have shown ambiguous behaviour. They have been tagged as ARG as well as PREP_M and SUBORD_A or SUBORD_M (adpositions), ADJ and ENTITIES (numerals) and ADJ (determiners). The class of ARG also contains articles. Finally, interjections have been classified as adverbs while EAGLES residual class is included in our entities class.

In the next section we describe our first experiments for the evaluation of the effects of the induced tagsets on the performances of automatic PoS-taggers.

---

[2]The sub-tag POLI was chosen because this is frequently the syntactic behaviour shown by Italian polysyllabic preposition.

4. Tagging experiments

In order to evaluate the effectiveness of the proposed PoS tagsets a number of experiments have been carried out. The two tagsets proposed in the previous section have been tested against the tagset proposed by TUT using three different taggers: (a) the CORISTagger, an HMM-based tagger which embodies a powerful Italian morphological analyser based on a 100.000-lemma lexicon (Tamburini, 2000), (b) the HMM-based tagger ACOPOST t3 (Schröder, 2002) and (c) the tagger SVMTool (Giménez and Màrquez, 2004) based on support vector machines. The two HMM-based taggers use a standard trigram model, while SVMTool allows for wider context and more sophisticated processing features (in our tests, features taken from a ±3-word context are considered and processed using a two-pass labelling method in a left-to-right and right-to-left fashion). The textual material contained in TUT has been extracted, annotated with the three different tagsets, split into a training set (33.414 words) and a test set (3720 words) and used to train and test the examined taggers. Table 2 summarises the three tested tagsets.

TUT	ADJ, ADV, ART, CONJ, DATE, INTERJ, NOUN, NUM, PHRAS, PREDET, PREP, PREP_A, PRON, PUNCT, VERB
TS1	ADJ, ARG_DET, ARG_PREP, ADV, COORD, ENTITIES, N , PREP_POLI, PREP_VM, PREP_NM, RELS, PUNT, SUBORD_A, SUBORD_M, V
TS2	ADJ, ARG_DET, ARG_PREP, ADV, COORD, ENTITIES, N, PREP_M, RELS, PUNT, SUBORD_A, SUBORD_M, V

Table 2: The three tagsets used in our experiments.

Table 3 outlines the results obtained in the tagging experiments considering a 'Baseline' experiment as well in which the most frequent tag for each word is selected.

	TUT	TS1	TS2
CORISTagger	94.36%	91.69%	92.58%
ACOPOST t3	93.21%	89.92%	91.08%
SVMTools	94.07%	89.44%	91.32%
Baseline	91.02%	86.13%	87.12%

Table 3: PoS tagging accuracy of the experimented taggers for the considered tagsets.

The tagging accuracy for all the performed experiments is quite low when compared with state-of-the-art results, but we have to consider that absolute performances are of no interest for this study; we are interested in investigating the relative performances between TUT tagset and our proposed tagsets. The training set we used in the experiments is extremely small when compared with the ones used in state-of-the-art experiments, which almost always contain some hundred-thousand words. This is the main explanation of the differences in tagging accuracy with other results.

The best performances, both as absolute values and relative ratio between tagsets, are obtained by the CORISTagger. The use of a powerful morphological analyser, able to successfully recover the common cases of unknown words, is likely to explain the difference.

However, there is a difference of about 2% between the tagging accuracy we obtained using the TUT tagset when compared both with TS1 and TS2 that requires some further comments. The PoS classes induced by the proposed method tend to describe relations that connect words or constituents that can be quite far from each other (long-distance dependencies) and, as outlined before, contain a rich set of prepositionals and subordinators classes. Limited context methods, such as the HMM tagging schemas considered here, are intrinsically unable to successfully manage such kind of relations. A careful evaluation of the tagging errors showed in table 4 highlights that the main source of dissimilarities between the performances obtained using these three tagsets is the different treatment of prepositions, as described in the previous sections.

TUT TS	TS1	TS2
39 NOUN-ADJ	64 PREP_VM-ARG_PREP	103 PREP_M-ARG_PREP
24 VERB-NOUN	44 PREP_VM-PREP_NM	30 N-ADJ
19 PRON-CONJ	30 PREP_NM-ARG_PREP	19 V-N
19 PREP-NOUN	30 N-ADJ	17 V-ADJ
18 VERB-ADJ	19 V-N	12 ARG_DET-ADJ
14 VERB-PRON	15 V-ADJ	10 ENTITIES-ADJ
11 PREP-ADV	11 ARG_DET-ADJ	7 SUBORD_M-SUBORD_A
10 PREP-CONJ	10 ENTITIES-ADJ	7 SUBORD_A-RELS
8 PRON-ADJ	8 SUBORD_A-RELS	6 N-ENTITIES
8 NOUN-ADV	7 SUBORD_M-SUBORD_A	6 N-ADV
6 NOUN-CONJ	6 N-ADV	5 SUBORD_A-PREP_M
6 ADV-ADJ	5 N-ENTITIES	5 PREP_M-ADJ
5 PREP_A-ADV	4 SUBORD_A-PREP_VM	4 SUBORD_M-PREP_M
4 PRON-NOUN	4 SUBORD_A-PREP_NM	4 ADV-ADJ
4 PREP_A-ART	4 PREP_VM-ADJ	
4 CONJ-ADV	4 ENTITIES-ARG_DET	

Table 4: Main PoS tagging errors of the CORISTagger for the considered tagsets. For each pair of tags the number of times in which the tagger confused them misclassifying a word is indicated.

In Italian, prepositions are involved in a wide range of highly specific syntactic constructions. As a result, the proposed tagset contains a number of PoS tags (ARG_PREP, PREP_POLI, PREP_NM, PREP_VM, SUBORD_M) encoding different and specific prepositional syntactic patterns, especially for the most frequent prepositions. From a lexical point of view, prepositions will receive all the possible tags, leading to highly ambiguous assignments, even in the TS2 case where the prepositional classes have been reduced.

5. Conclusions

This paper has presented a preliminary study on the induction of word classes (or PoS tags) starting from loosely labelled dependency structures encoding basic syntactic relations among words derived from an Italian treebank (TUT). Two slightly different tagsets have been induced and extensively tested using different state-of-the-art PoS taggers and the results have been compared with the ones obtained using a EAGLES conforming tagset (namely the TUT one). As a tendency we can observe that the design of a more informative and functionally oriented tagset leads to a performance lowering when using it in conjunction with standard stochastically-based tagging methods. A trade off has to be chosen between the opposite requirements of having

an informative tagset and accurate automatic tagging procedures.

An alternative approach to the problem could involve the development of different tagging techniques able to manage long-distance dependencies as usual parsing techniques can do. The results presented are derived from a preliminary study based on standard tagging techniques: our team is currently developing a tagging method that uses the word types derived directly from the type resolution phase in the tagging process, taking advantage of richer information in the tagging/parsing process.

6. References

S. Bangalore and A. Joshi. 1999. Supertagging: An approach to Almost Parsing. *Computational Linguistics*, 25(2):237–265.

R. Bernardi, A. Bolognesi, C. Seidenari, and F. Tamburini. 2005. Automatic induction of a POS tagset for Italian. In *Proc. Australasian Language Technology Workshop - ALTW 2005*, Sydney.

C. Bosco, V. Lombardo, Vassallo D., and Lesmo L. 2000. Building a treebank for Italian: a data-driven annotation schema. In *Proc. 2nd International Conference on Language Resources and Evaluation - LREC 2000*, pages 99–105, Athens.

C. Bosco. 2003. *A grammatical relation system for treebank annotation*. Ph.D. thesis, Computer Science Department, Turin University.

E. Brill and M. Marcus. 1992. Tagging an unfamiliar text with minimal human supervision. In *Proceedings of the Fall Symposium on Probabilistic Approaches to Natural Language*, pages 10–16, Cambridge.

A. Clark. 2000. Inducing Syntactic Categories by Context Distribution Clustering. In *Proceedings of CoNLL-2000 and LLL-2000 Conference*, pages 94–91, Lisbon, Portugal.

R. Delmonte. 2004. Strutture sintattiche dall'analisi computazionale di corpora di italiano. In A. Cardinaletti and F. Frasnedi, editors, *Intorno all'italiano contemporaneo. Tra linguistica e didattica*. Milano: F. Angeli.

J. Giménez and L. Màrquez. 2004. SVMTool: A general POS tagger generator based on Support Vector Machines. In *Proceedings of the 4th LREC*, pages 43–46, Lisbon.

F. Gobet and J. Pine. 1997. Modelling the acquisition of syntactic categories. In *Proceedings of the 19th Annual Meeting of the Cognitive Science Society*, pages 265–270.

M. Monachini. 1995. ELM-IT: An Italian Incarnation of the EAGLES-TS. Definition of Lexicon Specification and Classification Guidelines. Technical report, Pisa.

F. Pereira, T. Tishby, and L. Lee. 1993. Distributional clustering of English words. In *Proceedings of the 31st ACL*, pages 183–190, Columbus, Ohio.

M. Redington, N. Chater, and S. Finch. 1998. Distributional Information: a Powerful Cue for Acquiring Syntactic Categories. *Cognitive Science*, 22(4):425–469.

R. Rossini Favretti, F. Tamburini, and C. De Santis. 2002. CORIS/CODIS: A corpus of written Italian based on a defined and a dynamic model. In A. Wilson, P. Rayson, and T. McEnery, editors, *A Rainbow of Corpora: Corpus Linguistics and the Languages of the World*, pages 27–38. Munich: Lincom-Europa.

I. Schröder. 2002. A Case Study in Part-of-Speech tagging Using the ICOPOST Toolkit. *Technical report FBI-HH-M-314/02. Department of Computer Science, University of Hamburg*.

H. Schütze. 1993. Part-of-speech induction from scratch. In *Proceedings of the 31st ACL*, pages 251–258, Columbus, Ohio.

F. Tamburini, C. De Santis, and Zamuner E. 2002. Identifying phrasal connectives in Italian using quantitative methods. In S. Nuccorini, editor, *Phrases and Phraseology -Data and Description*, pages 45–64. Berlin: Peter Lang.

F. Tamburini. 2000. Annotazione grammaticale e lemmatizzazione di corpora in italiano. In R. Rossini Favretti, editor, *Linguistica e informatica: multimedialità, corpora e percorsi di apprendimento*, pages 57–73. Rome: Bulzoni.