

# A GENERATIVE GRAMMAR APPROACH FOR THE MORPHOLOGIC AND MORPHOSYNTACTIC ANALYSIS OF ITALIAN

Marina Russo

IBM Rome Scientific Center  
via del Giorgione, 129  
00147 Rome Italy

## ABSTRACT

A morphologic and morphosyntactic analyzer for the Italian language has been implemented in VM/Prolog [3] at the IBM Rome Scientific Center as part of a project on text understanding.

Aim of this project is the development of a prototype which analyzes short narrative texts (press agency news) and gives a formal representation of their "meaning" as a set of first order logic expressions. Question answering features are also provided.

The morphologic analyzer processes every word by means of a context free grammar, in order to obtain its morphologic and syntactic characteristics.

It also performs a morphosyntactic analysis to recognize fixed and variable sequences of words such as idioms, date expressions, compound tenses of verbs and comparative and superlative forms of adjectives.

The lexicon is stored in a relational data base under the control of SQL/DS [2], while the endings of the grammar are stored in the workspace as Prolog facts.

A friendly interface written in GDDM [1] allows the user to introduce on line the missing lemmata, in order to directly update the dictionary.

## Introduction

About thirty years ago, the development of deciphering techniques made computer scientists be involved for the first time in the field of Linguistics, especially in automatic translation matters.

The failure of most of these projects contributed to a general sensibilization towards natural language problems, and gave rise to a variety of formal theories for their treatment.

In the last few years, one of the main research objectives became the design of systems able to acquire knowledge directly from texts, using natural language as an interface between man and machine.

At the IBM Rome Scientific Center a system has been developed for processing Italian texts. The task of the system is to

- analyze short narrative texts (press agency news) on a restricted domain (Economics and Finance),
- give the formal representation of their "meaning" as a set of first order logic expressions, stored in a knowledge base,
- consult this knowledge base in order to answer any question about the contents of analyzed texts.

The system consists of:

- a **morphologic analyzer** based on a context-free logic grammar with the "word" as axiom and its possible components as terminal nodes. It includes a lexicon of about 7000 elementary

lemmata, structured in a table of a relational data base under the control of SQL/DS.

- a **morphosyntactic analyzer** realized by three regular grammars, recognizing respectively compound tenses of verbs (e.g. *has been signed*), comparative and superlative forms of adjectives (e.g. *the most interesting*) and compound numbers (e.g. *three billions 564 millions 234,000*). This module reduces the number of possible syntactic relations among the words of the sentence in order to simplify the task of the syntax.
- a **syntactic parser** developed by means of a meta-analyzer [6] which allows to write production rules for attribute grammars, and generates from these the corresponding top-down parser. A grammar has been written to describe the fragment of Italian considered.
- a **semantic processor** based on the Conceptual Graphs formalism [10] and provided with a semantic dictionary containing at present about 350 concepts. Its task is to solve syntactic ambiguities and recognize semantic relations between the words of the sentence [9].

This paper deals in particular with the structure of the lexicon adopted in the system and with the morphologic and morphosyntactic analyzer.

In this system the morphology and the lexicon are strictly combined; for this reason this lexicon does not contain semantic information. In the approach of Alinei [4], on the contrary, lexicon structures contain semantic information in order to describe every word also in terms of its "meaning".

Another possible approach is the one adopted by Zampolli who developed a frequency lexicon of Italian language at the Computational Linguistic Institute in Pisa [5]. The lexicon realized by Zampolli's working group contains morphologic hints in order to guide directly the analysis of every word, without the support of a morphologic parser.

In most of the works referring to English language morphology is considered only as a part of the syntactic parser. On the contrary, Italian morphology requires to be previously analyzed because it is more complex: there are more rules than in English and these rules present many exceptions.

For this reason, in the last few years Italian researchers began to face systematically these problems beside a purely linguistic context.

A procedural approach is the one followed by Stock in the development of a morphologic analyzer realized for the "Wednesday2" parser [11].

A different approach makes use of formal grammars to describe the rules of Italian morphology. This morphologic analyzer is based on a context free grammar describing the logic rules for the word generation. Other two morphologic systems have been developed according to the ATN formalism (Augmented Transition Network). The first one has been realized at the CNR Institute of Pisa by Morreale, Campagnola and Mugellesi, as a research tool for teaching Italian morphology, with applications in automatic processing of

natural language and knowledge representation [8]. The second one has been realized by Delmonte, Mian, Omologo and Satta, as part of a system for the development of a reading machine for blind people. [7].

In the first section of this paper there is a brief discussion about morphologic problems and about the possible approaches to their solution.

The next section describes the structure adopted for the lexicon and the other sets of data.

The third section deals with a preanalyzer, which simplifies the work of morphologic analysis by recognizing standard sequences of words, as idioms and date expressions.

In the fourth section the morphologic analyzer is described and in the last one the morphosyntactic analyzer, both realized by means of context free grammars.

## The problem

The aim of morphology is to retrieve from every analyzed word the lemma it derives from, its syntactic category (e.g. *verb*, *noun*, *adjective*, *conjunction*, ...) and its morphologic category (e.g. *masculine*, *singular*, *indicative*, ...).

A possible approach to the problem is to store in a data base a list of all the declined forms for every lemma of the language, as well as their morphologic, syntactic and semantic characteristics.

The size of such a list would be enormous, because a common dictionary contains about 50000-100000 lemmata and each lemma gives rise to several derived words and each word may be declined in different ways.

Such a large data base is hard to enter and to update, and it is limited by the fixed size of its words list.

In Italian, the creation of words is a generative process that follows several rules like, for instance:

	---	verbalization	---	MAN-EGGIARE (to hand-le)
MANO (hand)	---	composition	---	PALLA-MANO (hand-ball)
	---	cliticization	---	RI-MAN-EGGIARE (to re-hand-le)

In English, rules like composition or cliticization are not strictly morphologic, because they often involve more than a word. In Italian, on the contrary, they modify the single word, producing new words like, for instance:

	---	alteration	---	CART-ACCIA (waste paper)
CARTA (paper)	---	composition	---	CARTA-MONETA (paper money)
	---	cliticization	---	IN-CART-ARE (to wrap in paper)

These rules make the set of Italian words potentially unlimited, and sometimes make insufficient even a common dictionary.

A different approach takes two different lists: one containing the lemmata of the language and the other the logic rules of derivations, from which all the correct Italian words can be produced starting from the lemmata.

These rules can be easily described by means of a context-free grammar, in which every "word" results from the concatenation of the "stem" of a lemma with alterations, affixes, endings and enclitics.

This grammar can both *generate* from a given lemma all the current Italian words deriving from it and *analyze* a given word by

giving all the possible lemmata it derives from.

The backtracking mechanism of Prolog directly allows to obtain all the solutions.

This morphologic analyzer can also provide further information about some linguistic peculiarities, like, for instance:

<b>compound names</b>	<i>pelle-rossa</i> (red-skin), which has as plural <i>PELLI-ROSSE</i> .
<b>modal verbs</b>	which take another verb as object (I can go)
<b>altered names</b>	<i>foglia</i> (leaf) can be altered in <i>fogli-olina</i> (leaf-let), whose meaning is <i>piccola foglia</i> (small leaf).

## Data structure

A correct morphologic analysis requires not only knowledge on the language lemmata, but also on the word components as alterations, affixes, endings and enclitics. This information might be represented in form of Prolog facts. In this way, data might be directly accessed by the program, because the homogeneity of their structure. The disadvantage is a performance degradation when the size of data increases, since Prolog is not provided with efficient search algorithms.

Hence it seemed convenient to draw a distinction between data: on one hand the set of lemmata, and on the other the sets of affixes, alterations, endings and enclitics. The former (which is the most relevant and needs to be continuously updated), has been structured as a relational data base table, managed by the SQL/DS. The advantage is that this system is directly accessible from VM/Prolog (the string containing the query is processed by SQL, which returns the answer as a Prolog list). The latter (which have fixed length and are not so large), have been stored in the Prolog workspace in form of Prolog facts.

The set of lemmata is a table with five attributes:

1. the first is the lemma.
2. the second is the stem (the invariable part of the lemma): this is the access key in the table.
3. the third is the name of the "class of endings" associated with every lemma. A class of endings is the set of all the endings related to a given class of words. For example, each of the regular verbs of the first conjugation has the same endings; hence there exists a class named *dv1\_conjug* containing all and only these endings. Generally each irregular verb is related to different classes of endings: *andare* (to go), for example, admits two different stems, *vad* (go) and *and* (went); so there exist two subclasses of endings named respectively *dv1\_andare* and *dv2\_andare*.
4. the fourth attribute is the syntactic category of the lemma: for example, the information that *to have* is an auxiliary transitive verb.
5. the fifth is an integer identifying the type of analysis to be performed:
  - 1 the analysis can be performed completely
  - 2 the lemma can neither be altered nor affixed (this is the case for example of prepositions and conjunctions)
  - 3 only the longest analysis of the lemma is considered (this is the case of the false altered nouns: *mattino* (morning) is not a little *matto* (mad), such as in english *outlet* is not a little *out!*)

lemma	stem	ending_class	synt_cat	label
matto	matt	da_bello	adj.qualific.	1
mattino	mattin	dn_oggetto	noun.common	3
di	di	---	prep.simple	2
andare	vad	dv1_andare	v.intran.simple	1
andare	and	dv2_andare	v.intran.simple	1

The other sets of data are contained in the Prolog workspace and are structured as tables of a relational data base.

The set of the classes of endings is a table with three attributes:

1. the first is the name of the class and it is the access key in the table.
2. the second is one of the endings belonging to the class
3. the third is the morphologic category associated with the ending: for example, the class *dn\_oggetto* contains the two endings which are used in order to inflect all the masculine nouns behaving like the word *oggetto* (object): *o* for the singular (*oggett-o*), and *i* for the plural (*oggett-i*).

ending_class	ending	morph_cat
dn_oggetto	o	mas.sing.
dn_oggetto	i	mas.plur.

The affixes can be divided in **prefixes** preceding the stem of the lemma, and **suffixes** following the stem of the lemma.

The prefixes are simply listed by means of a one attribute table. In this way it is not necessary to list the prefixed words in the lexicon: they are obtained by chaining the prefix with the original word. For example, from the verb *to handle* with the prefix *re* we obtain the verb *to rehandle*. Morphologic and syntactic characteristics remain the same; for the verbs only, the prefixed verb differs sometimes from the previous one in the syntactic attributes (transitive/intransitive, simple/modal).

The set of suffixes is a table with four attributes:

1. the first is the suffix itself
2. the second is the stem of the suffix (the access key to the table)
3. the third is the ending class of the suffix
4. the fourth is the syntactic class of the suffix. Suffixes, in fact, differently from prefixes, changes both morphologic and syntactic characteristics of the original word: they change verbs into names or adjectives (*verbal suffixes*), names into verbs or adjectives (*denominal suffixes*), adjectives into verbs or names (*deadjectival suffixes*). The first attribute is chained to the stem of the original lemma in order to obtain the derived lemma: for example, from the stem of the lemma *mattino* (morning), which is a noun, with the suffix *iero*, we obtain the new lemma *mattin-iero* (early rising), which is an adjective, and from the second stem of the lemma *andare* (to go), which is a verb, with the suffix *amento*, we obtain the new lemma *and-amento* (walking), which is a noun.

suffix	stem	ending_class	synt_cat
iero	ier	da_bello	adj.qualific.
amento	ament	dn_oggetto	noun.common

The set of alteration is a table with three attributes:

1. the first is the stem of the alteration (the access key in the table)
2. the second is the ending class of the alteration
3. the third is the semantic type of the alteration. Alterations

change the morphologic and semantic characteristics of the altered word, but not its syntactic category: for example, the lemma *casa* (*house*) can be altered in *casina* (*little house*), *casona* (*big house*), *casaccia* (*ugly house*), and so on:

stem	ending_class	seman_cat
in	da_bello	diminutive
on	dn_cosa	augmentative
acc	da_anfibio	pejorative

The enclitics are pronouns linked to the ending of a verb: for example *va li'* (*go there*) can be expressed also in the form *vacci* (*ci* is the enclitic, the *c* is duplicated according with a phonetic rule).

The set of the enclitics is a table with two attributes: the first is the enclitic (this is the access key to the table) and the second is the morphologic characteristic of the enclitic. The analyzer divides the verb from the enclitic, so that it becomes a different word, taking the morphologic characteristic stated in the table and the syntactic category of *pronoun*.

Other two sets of data have been defined in order to handle fixed sequences of words, such as proper names and idioms.

The set of the most common italian idioms has been structured as a table with two attributes: the first one is the idiom itself, while the second is the syntactic category of the idiom. In this way it is possible to recognize the idiom without performing the analysis of each of the component words. For example, *di modo che* (in such a way as) is an idiom used in the role of a conjunction, and *a mano a mano* (little by little) is used in the role of an adverb.

The set of proper names belonging to the context of Economics and Finance is a table with three attributes: the first is the proper name, the second its syntactic category and the third its morphologic category.

proper_name	synt_cat	morph_cat
lunedì' ( <i>monday</i> )	name.prop.wday	mas.sing.
Montopolimeri Montedison	name.prop.comp.	fem.sing.
Vittorio Ripa di Meana	name.prop.pers.	mas.sing.
Reggio Emilia	name.prop.loc.	fem.sing.

## The Preanalyzer

The preanalyzer simplifies the work of analysis recognizing all the "fixed" sequences of words in the sentence.

Fixed sequences of words are, for example, idioms like *in such a way as*. To analyze this sequence of words it is not necessary to know that *in* is a preposition, *such* is an adjective, *a* an article, and so on: the only useful information is that this sequence takes the role of conjunction. Other fixed sequences of words are proper names: it is necessary to know, for example, that *Montopolimeri Montedison* or *Vittorio Ripa di Meana* are single entities.

Idioms and proper names are recognized by means of a pattern matching algorithm: the comparison is made between the input sentence and the first attribute of the tables of idioms and proper names. When the comparison fails, backtracking evaluates another hypothesis. Every recognized sequence of words is written on an appropriate file and then removed from the input sentence.

Date expressions, as *lunedì' 13 agosto* (*monday, august the 13rd*), are considered as single entities, in order to simplify the work of syntax. They are recognized by means of a context-free grammar,

whose axiom is the 'date':

```

1 DATE --> <name_proper_wday> <DA1>
2 DATE --> <DA1>
3 DATE --> <DA2>
4 DA1 --> <number(<31>> <name_proper_month>
5 DA1 --> <number(<31>> <DA2>
6 DA2 --> <name_proper_month> <number>

```

Figure 1. The grammar for the DATE

Numbers are recognized by the library function *numb(\*)* and by means of a context-free grammar translating strings into numbers. In this way it is possible to evaluate in the same way expressions such as 1352 and milletrecentocinquantadue (one thousand three hundred and fifty two).

```

1 NUMBER --> <NUM1>
2 NUMBER --> <'mille'>
3 NUMBER --> <'mille'> <NUM1>
4 NUMBER --> <NUM1> <'mila'>
5 NUMBER --> <NUM1> <'mila'> <NUM1>
6 NUM1 --> <NUM2>
7 NUM1 --> <NUM3>
8 NUM1 --> <NUM4>
9 NUM2 --> <units> <NUM3>
10 NUM3 --> <'cento'>
11 NUM3 --> <'cento'> <NUM4>
12 NUM4 --> <units>
13 NUM4 --> <tens>
14 NUM4 --> <tens> <units>

```

Figure 2. The grammar for the NUMBER

## The morphologic analyzer

This is the main module of the whole system. Its task is to analyse each element (word) of the list received from the preanalyser and to produce for every form analyzed the list of all its characteristics:

1. the lemma it derives from
2. its syntactic characteristics
3. its morphologic characteristics (none for invariable words)
4. the list of alterations (possibly empty)
5. the list of enclitics (possibly empty).

For example the form *sono* (the 1st sing. and the 3rd plur. person of the present indicative of *essere*, *to be*), after the analysis is represented by the list:

```

(sono.
 (v.intran.aux.ind.pres.act.1.sing.essere.nil).
 (v.intran.aux.ind.pres.act.3.plur.essere.nil).
 nil)

```

Every Italian word is made up by a fundamental nucleus, the *stem* (two for the compound names). This is preceded by one or more *prefixes*, and followed by one or more *suffixes* and *alterations*, by an *ending* and, as far as the verbs are concerned, by one or more *enclitics*.

This structure has been described by means of a context-free grammar in which the "word" is the axiom and all its components the endings.

```

1 WORD --> {prefix}n <stem> <REM>
2 REM --> {suffix}n {alteration}n <TAIL>
3 REM --> <ending> {suffix}n {alteration}n <TAIL>
4 TAIL --> <ending> {enclitic}n

```

Figure 3. The grammar for the WORD

Here are some example of words analyzed with this grammar:

- *muraglione* (high wall)
  - mur** is the stem of the word *muro* (wall)
  - agl** is the stem of the suffix *aglia*
  - i-on** *on* is the stem of the alteration *one* (augmentative); the *i* is an euphonic vowel
  - e** is the ending of the singular.

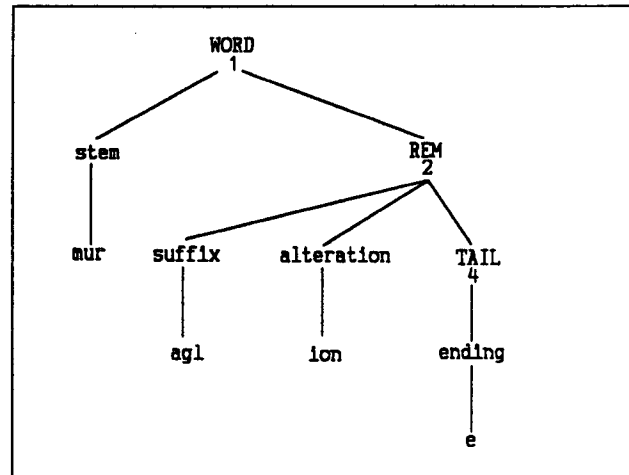


Figure 4. Parse tree for the word MURAGLIONE

- *trasportatore* (carrier)
  - tras** is the prefix
  - port** is the stem of the verb *portare* (to carry)
  - at** is the ending of the past participle of the verb
  - or** is the stem of the deverbal suffix *ore*
  - e** is the ending of the masculine singular.

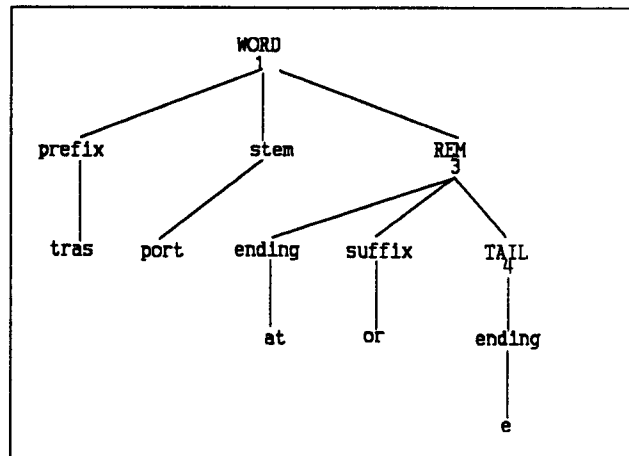


Figure 5. Parse tree for the word TRASPORTATORE

- **ridandoglielo** (giving it to him/her again)
  - ri** is the prefix (it means *again*)
  - d** is the stem of the verb *dare* (to give)
  - ando** is the ending of the present tense of gerund of the verb
  - glie** is the first enclitic (it means *to him/her*): *e* is an euphonic vowel
  - lo** is the second enclitic (it means *it*).

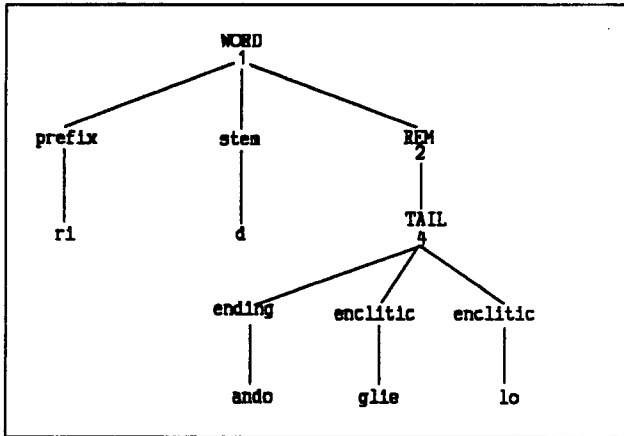


Figure 6. Parse tree for the word RIDANDOGLIELO

The compound nouns are not reported in the lexicon: they are derived from the two component lemmata. Their plural is made according to the following set of rules:

1	V + N(mas.sing)	--> Noun's ending changes
2	V + N(fem.sing)	--> no ending changes
3	V + N(plur)	--> no ending changes
4	V + V	--> no ending changes
5	N + N	--> 2nd Noun's ending changes
6	Adj + N	--> Noun's ending changes
7	N + Adj	--> both endings change

Figure 7. The rules for the plural of Compound Nouns

Some examples of compound nouns are:

rule	singular	plural
1	passa-porto ( <i>pass-port</i> )	passa-porti
2	porta-cenere ( <i>ash-tray</i> )	porta-cenere
3	cava-tappi ( <i>cork-screw</i> )	cava-tappi
4	sali-scendi ( <i>door-latch</i> )	sali-scendi
5	banco-nota ( <i>bank-note</i> )	banco-note
6	basso-rilievo ( <i>bas-relief</i> )	basso-rilievi
7	cassa-forte ( <i>steel-safe</i> )	casse-forti

The task of this part of the morphology is to:

1. **recognize** all the "well-formed" words of Italian language. The analyzer parses the words from left to right, splitting them into elementary parts: prefix(es), the stem(s) of the appropriate lemma(ta) of derivation (retrieved from a restricted dictionary reporting only the "elementary lemmata") suffix(es), alteration(s), ending(s), enclitic(s). Each hypothesis is checked by verifying that all the conditions for a right composition of those parts are satisfied.

2. **submit** every word not recognized to the user, who can state whether:
  - the word is really wrong, because of
    - an orthographic error: for example *squola* instead of *scuola* (school).
    - a composition error: for example *serviziazione* is wrong as 'iazione' is a deverbal suffix and 'serviz' is the stem of the noun 'servizio' (service) and the corresponding verb does not exist.
  - the word derives from a lemma which is not reported in the lexicon. In this case the user can recall a graphic interface, allowing him/her to update directly the lexicon.
3. **perform**, if requested by the user, an inspection in the list of the "currently used" words. In this way, for example, the user knows that *coton-ificio* (cotton-mill) and *coton-iera* are two well-formed Italian words, but that only the first one is commonly used.

## The morphosyntactic analyzer

The aim of the morphosyntactic analyzer is to perform the analysis of the contiguous words in the sentence, in order to recognize regular structures such as compound tenses of verbs and comparative and superlative forms of adjectives.

Compound tenses of verbs are described by means of a regular grammar, whose rules are applied any time the analyzer finds in the sentence the past participle of the verb. These rules are:

1	COMP_TENSE	--> <v.tran.aux.>
		<v.tran.(past.part.)>
2	COMP_TENSE	--> <v.intran.aux.> <REM>
3	REM	--> <v.intran.aux.(past.part.)>
		<v.tran.(past.part.)>
4	REM	--> <v.tran.(past.part.)>
5	REM	--> <v.intran.(past.part.)>

Figure 8. The grammar for the COMPOUND TENSES of verbs

When a rule is successfully applied the morphologic categories of the verbs are changed and the attribute 'active'/'passive' can be specified correctly. For example, after the morphosyntactic analysis, the phrase *io sono chiamato* (*I'm called*)

```
((io.
  (pron.pers.1.sing.io.nil).
  nil).
 (sono.
  (v.intran.aux.ind.pres.act.1.sing.essere.nil).
  (v.intran.aux.ind.pres.act.3.plur.essere.nil).
  nil).
 (chiamato.
  (v.tran.sim.part.past.act.mas.sing.chiamare.nil).
  nil).
 nil)
```

becomes

```
((io.
  (pron.pers.1.sing.io.nil).
  nil).
 (sono_chiamato.
  (v.tran.sim.pass.ind.pres.1.sing.chiamare.nil).
  nil).
 nil).
```

in which only the first analysis of the word "sono" has been taken, as the number of the auxiliary verb must correspond to the number of the past participle. The form is passive, as "chiamare" (*to call*) is a transitive verb (the auxiliary verb for the active form is *to have*). In

this case morphosyntactic analysis has solved an ambiguity: only an interpretation will be analyzed by syntax.

The following figure shows the task of the grammar, applied any time the parser finds the past participle of a verb in the sentence.

- If the verb is transitive the parser looks at the word BEFORE the verb:
  - if the word is a tense of the verb to be, the resulting verb is SIMPLE PASSIVE (the rules applied are the 2nd and the 4th);
  - if the word is a tense of the verb to have, the resulting verb is COMPOUND ACTIVE (the rule applied is the 1st).
- If the verb is intransitive the parser looks at the word AFTER the verb:
  - if it is the past participle of another verb the resulting verb is COMPOUND PASSIVE (the rules applied are the 2nd and the 3rd);
  - otherwise it is COMPOUND ACTIVE (the rules applied are the 2nd and the 5th).

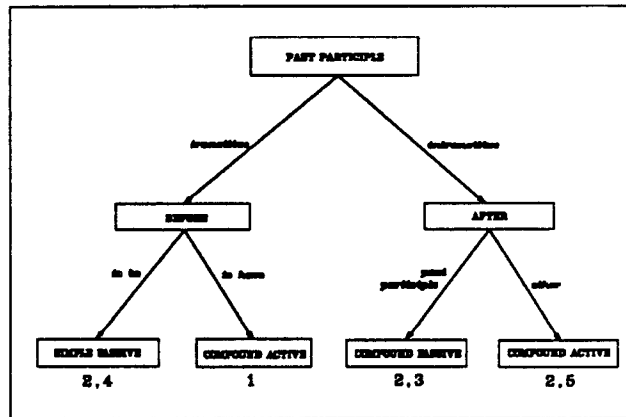


Figure 9. Compound tenses of verbs

The grammar for the comparative and superlative forms of adjectives is applied any time the analyzer finds the words *piu'* (*more*), *meno* (*less*) followed by a qualificative adjective. In this way it is possible to recognize and to distinguish expressions like *piu' interessante* (*more interesting*) and *il piu' interessante* (*the most interesting*). Remark that in English there is the use of *more*, *most* to make clear the distinction between the comparative and the superlative form of the adjective.

1	SUPERL REL	-->	<art.determ.>	<COMPARATIVE>
2	COMPARATIVE	-->	<'piu''>	<adj.qualific.>
3	COMPARATIVE	-->	<'meno''>	<adj.qualific.>

Figure 10. The grammar for the SUPERLATIVE and COMPARATIVE form of adjectives

In the same manner it is possible to recognize mixed numeric expressions like *three billions 564 millions 234000* and to evaluate them into their equivalent numeric form (*3564234000*). The rules are applied any time the analyzer finds the words *miliardi* (*billions*), *milioni* (*millions*) in the sentence.

1	NUM_COMP	-->	<agg.num>	<'miliardo''>	<NUM1>
2	NUM_COMP	-->	<agg.num>	<'miliardo''>	<agg.num>
3	NUM_COMP	-->	<agg.num>	<'miliardo''>	
4	NUM_COMP	-->	<NUM1>		
5	NUM1	-->	<agg.num>	<'mилione''>	<agg.num>
6	NUM1	-->	<agg.num>	<'mилione''>	

Figure 11. The grammar for COMPOUND NUMBERS

## Conclusions

This approach presents the advantage of a higher flexibility in the analysis of words. Moreover such a method has requested a strong initial effort in the formalization of the rules (with all their exceptions) for the morphologic treatment of words, but has largely simplified the work of classification of every Italian word.

The lexicon stores about 7000 elementary lemmata, derived from a list of about 20000 different Italian forms. They correspond to about 15000 ordinary lemmata (entries of a common dictionary).

## References

- [1] **Graphical Data Display Manager**, *Application Programming Guide*, SC33-0148-2, IBM Corp., 1984.
- [2] **SQL/Data System**, *Terminal User's Reference*, S1124-5017-2, IBM Corp., 1983.
- [3] **VM/Programming in Logic**, *Program Description/Operation Manual*, SH20-6541-0, IBM Corp., 1985.
- [4] M.Alinei, *La struttura del lessico*, ed. Il Mulino, 1974.
- [5] U.Bortolini, C.Tagliavini and A.Zampolli, *Lessico di frequenza della lingua italiana contemporanea*, ed. IBM, 1971.
- [6] B.Bottini and M.Cappelli, *Un Meta Analizzatore Orientato al Linguaggio Naturale in Ambiente Prolog*, *M.D. Thesis*, Milano, 1985.
- [7] R.Delmonte, G.A.Mian, M.Omologo and G.Satta, *Un riconoscitore morfologico a transizioni aumentate*, *Proceedings of AICA Meeting*, Florence, 1985.
- [8] E.Morreale, P.Campagnola and R.Mugellesi, *Un sistema interattivo per il trattamento morfologico di parole italiane*, *Proceedings of AICA Meeting*, Pavia, 1981.
- [9] M.T.Pazienza and P.Velardi, *Pragmatic Knowledge on Word Uses for Semantic Analysis of Texts*, *Workshop on Conceptual Graphs*, Thornwood, NY, August 18-20 1986.
- [10] J.F.Sowa, *Conceptual Structures: Information Processing in Mind and Machine*, *Addison-Wesley*, Reading, 1984.
- [11] O.Stock, F.Ceccconi and C.Castelfranchi, *Analisi morfologica integrata in un parser a conoscenze linguistiche distribuite*, *Proceedings of AICA Meeting*, Palermo, 1986.