A Greek Morphological Lexicon and its Exploitation by Natural Language Processing Applications

Georgios Petasis, Vangelis Karkaletsis, Dimitra Farmakiotou,
Ion Androutsopoulos*, and Constantine D. Spyropoulos

Software and Knowledge Engineering Laboratory
Institute of Informatics and Telecommunications
National Centre for Scientific Research "Demokritos"
Aghia Paraskevi, Athens, 153 10 Greece
{petasis, vangelis, dfarmak, ionandr, costass}@iit.demokritos.gr

Abstract. This paper presents a large-scale Greek morphological lexicon, developed at the Software & Knowledge Engineering Laboratory (SKEL) of NCSR "Demokritos". The paper describes the lexicon architecture and the procedure to develop and update it. The morphological lexicon was used to develop a lemmatiser and a morphological analyser that were exploited in various natural language processing applications for Greek. The paper presents these applications (controlled language checker, information extraction, information filtering) and discusses further research issues and how we plan to address them.

1 Introduction

During the last decade, we have witnessed a remarkable acceleration in the growth of Internet, communication networks, multimedia, etc. In this new era, the main vehicle for digital content products and services is natural language, increasing the need for robust language engineering systems. Language resources, such as lexicons and grammars, constitute the main ingredient of such systems. For this reason, there is a strong need for development of language resources that can be exploited by various natural language processing applications. For instance, lexicons with morphological and syntactic information are needed for the development of tools such as spelling and syntax checkers that can be integrated in word processors, as well as for the development of morphological and syntactic analysers that can be exploited by more complex natural language processing applications (search engines, information filtering and extraction systems, machine translation systems, etc.).

The Greek institutions involved in language engineering have paid special attention to the development of Greek language resources. During the last few

* Now with the Department of Informatics, Athens University of Economics and Business, Greece. E-mail: ion@aueb.gr

years, computational lexicons for Modern Greek have been developed by the Computer Technology Institute, the Institute for Language & Speech Processing, the Wire Communications Laboratory at the University of Patras, and the Software & Knowledge Engineering Laboratory at NCSR "Demokritos". The development of computational lexicons (i.e. lexicons that can be exploited by natural language processing applications) is a difficult task and becomes even more difficult due to the characteristics of the Modern Greek language. The complexity of the Modern Greek inflectional system, the existence of marked stress, free-word-order, the parallel use of Ancient Greek word forms and inflections along with Modern Greek word forms and inflections, as well as use of foreign words that have been partly incorporated or have not been incorporated in the Modern Greek language system are the main characteristics of Modern Greek that affect the computational treatment of its morphology.

The lexicon of the Computer Technology Institute (CTI) contains ∼80.000 lemmas (∼1.000.000 word-forms) [7]. Given a word-form, the CTI lexicon returns the corresponding lemma (or lemmas in case of lexical ambiguity) along with morphosyntactic and semantic information, i.e. part of speech, number, gender, case, person, tense, voice, mood, etc. The CTI lexicon is based on the CTI lexicon formalism for the description of inflected words. The CTI formalism treats each stem as a lexicon entry, embodies rules for inflectional morphology and marked stress, and denotes morphosyntactic or semantic attributes of morphemes and syntactic or semantic relations involving lemmas. The CTI lexicon has been used as the basis for the Greek spelling checker adopted by Microsoft for its word-processor MS Word. This lexicon has also been used recently for the development of the Greek WordNet, a semantic network that includes for every lemma not only morphosyntactic but also semantic information (synonyms, semantic groups-synsets, etc.) based on the EuroWordnet formalism (project EPET-II DIALEXICO).

The lexicon of the Institute of Language & Speech Processing (ILSP) has been developed in the context of the EC project LE-PAROLE, aiming at natural language processing applications. It contains ∼20.000 lemmas encoded at the morphological and the syntactic level according to the PAROLE/SIMPLE schema. Each lemma is represented as a Morphological Unit, which may correspond to more than one Graphical Morphological Units expressing in this manner alternative spellings of same lemma and is linked to Combinations of Morphological Features (number, gender etc.). Each Graphical Morphological Unit is linked to a Graphical Inflectional Paradigm, which expresses how different word forms are generated from the lemma. The initial ILSP lexicon has been extended in the context of the EPET-II project LEXIS. The new version of the lexicon is comprised of ∼65.000 lemmas containing morphological information of which a subset also contains syntactic and semantic information [1].

The lexicon of the Wire Communications Laboratory (WCL) contains ∼35.000 lemmas along with the inflected forms of the words and their grammatical features stored in a Directed Acyclic Word Graph (DAWG). This lexicon was exploited in the context of the EPET-II project MITOS for the development

of a fast morphological analyser [9]. The morphological analyser results are used by the MITOS[1] information extraction system.

The lexicon presented in this paper has been developed at the Software & Knowledge Engineering Laboratory (SKEL) of NCSR "Demokritos". The SKEL lexicon consists of ∼60.000 lemmas that correspond to ∼710.000 different word forms. The SKEL lexicon has been developed in parallel with *Ellogon* [8], a general-purpose text-engineering platform which facilitates the development of new tools as well as their integration in different applications. Thus the SKEL lexicon or its components can be easily embedded in different applications taking advantage of the facilities provided by *Ellogon*.

The SKEL lexicon architecture, the procedure to create it, and the provided functionalities for updating it, are presented in Section 2. The morphological lexicon has been exploited in the development a lemmatiser and a morphological analyser, which have been integrated in a controlled language checker for Greek, in Greek information extraction systems, as well as in a Greek information filtering system. Sections 3, 4 and 5 discuss the lexicon exploitation in the context of these natural language processing applications. Finally, section 6 presents further research issues and how we plan to address them. Our goal is to produce a wide-coverage morphological lexicon of Modern Greek that can be easily maintained and that can be easily exploited in new natural language processing applications.

2 The SKEL Lexicon for Modern Greek

In this section we describe the lexicon architecture and organisation, the way it was originally created and the infrastructure provided for accessing and maintaining its morphological database.

2.1 Lexicon Organisation

The lexicon consists of two independent components, the *query component* and the *generation component*. The query component is responsible for querying the lexicon about a specific word form and retrieving the associated linguistic information of a word form (Fig. 1). The query component is organised around a morphological database, which associates word forms with sets of morphological entries. Morphological entries are the basic elements for storing morphological information. Each morphological entry contains a fixed number of fields describing a specific word form, where each field represents a morphological feature, such as the lemma or the part of speech of the word form. A complete list of available fields as well as all their corresponding values is presented in Table 1.

In Greek it is often the case that the same word form may be found in text associated with different sets of morphological features. Thus, more than one morphological entry may be associated with a single word form. Figure

[1] http://www.iit.demokritos.gr/skel/mitos/

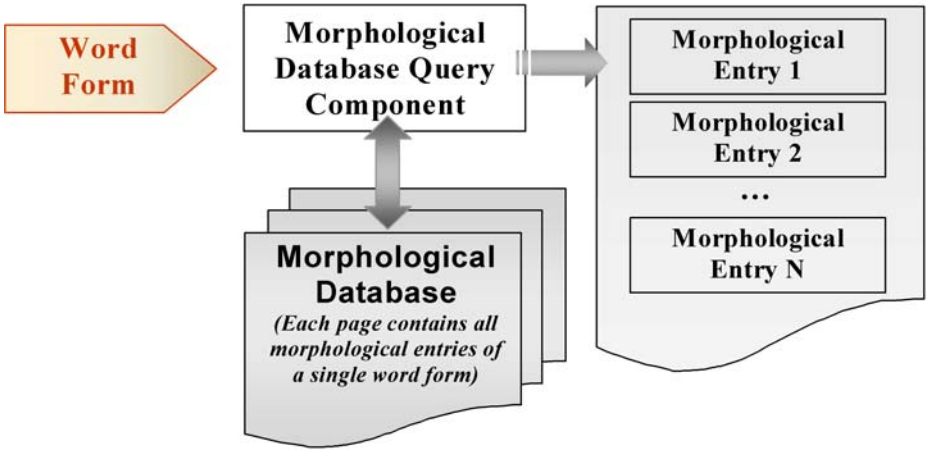

Fig. 1. The lexicon Query Component.

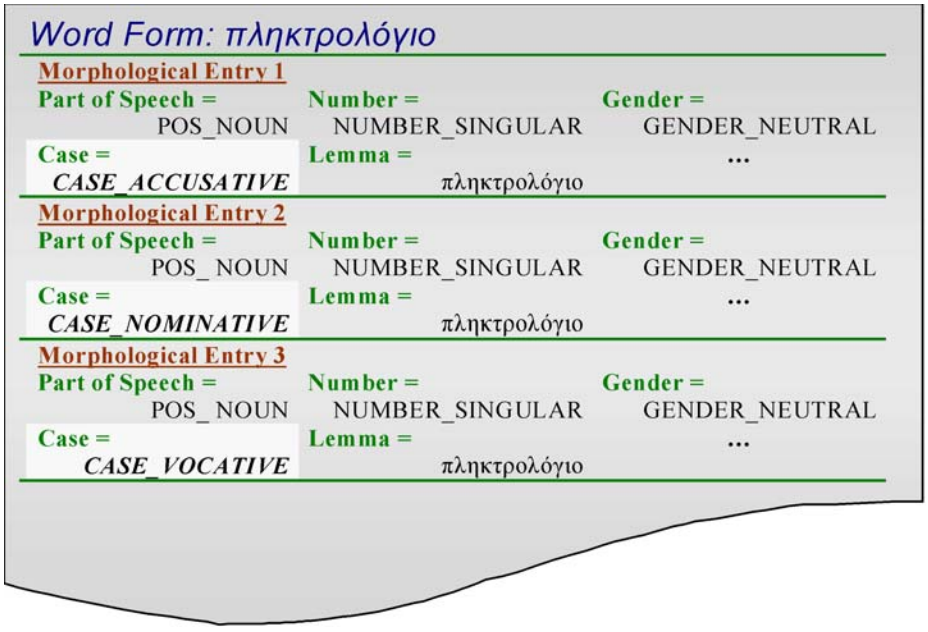

Fig. 2. A page from the morphological database, describing the word form "πληκτρολόγιο".

2 for example, presents all the morphological entries associated with the word form "πληκτρολόγιο" (keyboard): all three morphological entries share the same values for all features (i.e. the same lemma, part-of-speech, number, etc.), except

the case feature, as the same word form can appear in texts having three different case values. All morphological entries associated with the same word form are regarded as entries belonging to the same *page*.

The morphological database comprises of a fixed number of *pages*. As each page is associated with a unique word form, there are as many pages as the number of different word forms the lexicon can recognise. The number of morphological entries in each page is equal to the number of all the different instantiations of a word form the lexicon is aware of. During a word form search, the query component locates the page that describes the requested word form. If such a page is located (i.e. if the word form is contained in the database), all its morphological entries are returned (Fig. 2).

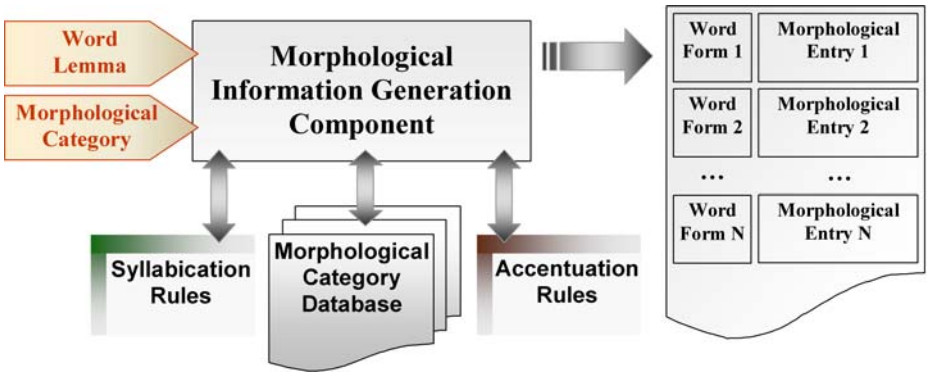

Fig. 3. The lexicon Generation Component.

The generation lexicon component is responsible for generating all the possible word forms as well as all the morphological entries of each word form for a given lemma. Apart from the lemma, this component also requires classification of the given lemma in one of the predefined morphological categories contained in the morphological categories database (Fig. 3). Each morphological category contains instructions describing how the various word forms can be generated from the word lemma and what morphological feature values must be associated with it. An example is presented in Table 2: this category can be used to create the instantiations of male proper names ending in "ός", like the name "Ασκληπιός". Given the word lemma and an appropriate morphological category, the generation component also utilises language specific rules regarding syllabication and accentuation in order to produce all possible word forms. During the creation process, each generated word form is represented with one morphological entry. As a result, a word form can be generated more than once if ambiguity exists, but each instance will be represented by a different morphological entry. For example, if we had the word lemma "πληκτρολόγιο" the word form "πληκτρολόγιο" would be generated three times (Fig. 2). However, if all

morphological entries for a generated word form are collected, the resulting set may not in general be a page, as a specific word form may also be generated by other lemmas that may even belong to different parts of speech. For example many articles (e.g. "$\tau o \upsilon$") share the same word forms with pronouns. If the generation component is used on the article lemma ("o") the returned morphological entries for the word form "$\tau o \upsilon$" will not be a page, as the morphological entries regarding its instantiations as a personal pronoun will be missing.

	Morphological Entry Field	Available Field Values
Morphological Entry	Word Form	The word form
	Lemma	The word lemma
	Stem	The word stem
	Suffix	The word suffix
	Part of Speech	POS_ARTICLE, POS_NOUN, POS_ADJECTIVE, POS_PRONOUN, POS_VERB, POS_PARTICIPLE, POS_ADVERB, POS_PREPOSITION, POS_CONJUNCTION, POS_PARTICLE
	Number	NUMBER_SINGULAR, NUMBER_PLURAL
	Case	CASE_NOMINATIVE, CASE_GENITIVE, CASE_DATIVE, CASE_ACCUSATIVE, CASE_VOCATIVE
	Tense	TENSE_PRESENT, TENSE_PAST_CONTINUOUS, TENSE_FUTURE_CONTINUOUS, TENSE_FUTURE, TENSE_PAST, TENSE_PRESENT_PERFECT, TENSE_PAST_PERFECT, TENSE_FUTURE_PERFECT
	Translation	An English translation, if available
	Other Fields	Info, Mood, Mode, Voice, Person, Syllabication, Part of Speech Detail, Inflectional Type, Accented Syllable, Gender, Inflection, Explanation, Examples, Synonyms

Table 1. Morphological Entry fields and their permissible values.

2.2 Lexicon Creation

Once the infrastructure described above was available, an initial version of the lexicon was created. Initially, a list of word lemmas was constructed. In order to collect as many word lemmas as possible, various textual corpora have been used, as well as freely available lists of words intended to be used by Greek versions of open source spell checkers (like "ispell" and "aspell"). The list of word forms collected from all these sources contain approximately 260.000 unique word forms. These word forms were examined in order to identify and fix errors as well as to extract the corresponding word lemmas. Finally, the list of word lemmas was enriched with proper names (names of persons and locations) that were extracted from the various lists of proper names (gazetteers) developed at our laboratory. Currently, the list of word lemmas contains approximately 60.000 unique lemmas.

Category Type	PNM_1		
Suffix	ός		
Part of speech	Noun		
Inflectional type	ACCENT_OXYTONO		
Inflection	INFLECTION_EQSYL		
Info	Proper Noun		
Generative Suffix	**Case**	**Number**	**Accented Syllable**
-ός	CASE_ACCUSATIVE	NUMBER_SINGULAR	1
-ού	CASE_GENITIVE	NUMBER_SINGULAR	1
-ό	CASE_NOMINATIVE	NUMBER_SINGULAR	1
-έ	CASE_VOCATIVE	NUMBER_SINGULAR	1

Table 2. A Morphological category example.

As a next step, the listed lemmas were classified in categories using simple heuristics. Word forms that shared the same stems with a lemma (based on heuristics to remove prefixes) were considered different instantiations of the lemma. Lemmas that share the same suffix and also their associated instantiations share common suffixes were classified in common categories. Finally, the morphological categories created with the automatic classification were reviewed by human experts and generation instructions were added for each category. This simple approach worked remarkably well for some parts of speech, like adjectives and nouns. However, it has not managed to generate categories for verbs, articles, pronouns and all the other parts of speech. The main reason for this was either the fact that these parts of speech are not inflected, or the fact that their inflected word forms were very diverse from the lemma or other word forms associated to the lemma. However, lemmas that belong to closed categories like articles or pronouns can be classified fast. On the other hand, lemmas for verbs are more difficult to classify into categories. For instance, past tenses are mostly characterised by prefixes or infixes added to the stem, besides the inflectional suffixes for different tenses. For example, the past form of "*κάν-ω*" (do) is "*έ-καν-α*". At the same time, the various participles are quite similar, even for verbs that belong to different declension categories, making their classification to different categories impossible with our simple heuristics, as the evidence is limited.

After the morphological categories had been manually corrected, word lemmas were manually classified according to their part of speech and morphological category. Approximately 350 morphological categories were created, covering mainly nouns, adjectives, verbs and pronouns. The number of morphological categories is not fixed since new categories may be added to cover new words. The process of manual classification of a word lemma into a morphological category is partially supported by a specialised tool that is able to propose possible morphological categories (Fig. 4). With this tool, the user can select any of the proposed categories and see all the word forms that can be generated if the word lemma is classified into the selected inflectional category. In case all proposed

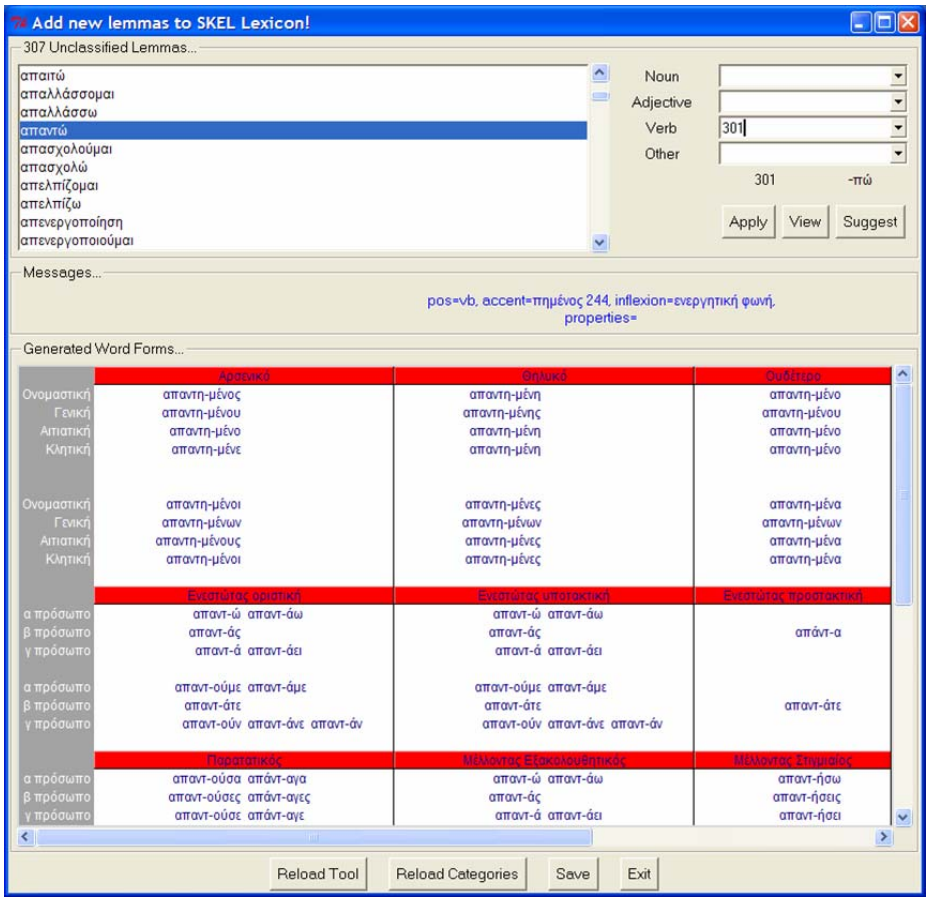

Fig. 4. A tool for updating the lexicon with new words.

morphological categories are inadequate, the user can create a new category and classify the word lemma in it.

The last step of the process was to process the morphologically classified word lemmas with the lexicon generation component. The generation component created all word forms as well as all relevant morphological entries for each word form and filled the morphological database of the lexicon. From the initial list of approximately 60.000 unique word lemmas, 710.000 different word forms were generated, leading to ∼2.500.000 morphological entries in the morphological database. Approximately 3.000 word lemmas were not processed by the generation component due to various errors (including errors in morphological category classifications detected by the generation component). In Table 3 the distribution of lemmas over parts of speech is shown, as well as the number of currently defined morphological categories.

Part of Speech	Lemma Number	Percentage (%)	Morphological Categories Number
Noun	29744	50,82	201
Adjective	13203	22,56	107
Verb	5850	10,00	48
Adverb	234	0,40	–
Other	9495	16,22	–

Table 3. The distribution of lemmas to the various parts of speech.

2.3 Lexicon Access and Maintenance

Both the query and the generation components as well as the whole software infrastructure of the lexicon have been developed in the C++ programming language, as our main concern was to build a portable and efficient system that could be easily embedded inside other applications that need to access the lexicon. This infrastructure offers an object-oriented environment that facilitates memory management and allows the insertion of an abstraction layer between the lexicon functionality and the specific internal details of the lexicon implementation. Through the provided programming interface (API) the caller can access both the query and generation components. Additionally, the software offers direct access to the morphological database by offering the ability to insert new morphological entries as well as to retrieve, modify or delete existing ones. Having direct access to the morphological entries of the database, the caller can extract part of the information contained in a morphological entry and create a separate, specialised database to satisfy specific needs. For example, one may extract a lemmatiser for specific purposes from the lexicon, e.g. a lemmatiser that associates word forms with the corresponding lemmas, ignoring all other pieces of information, resulting in a specialised tool that can be used independently of the lexicon.

The modularity and the provided API of the lexicon infrastructure have permitted the embedding of the lexicon infrastructure under the Tcl programming language. Tcl is an easy to learn, high level scripting language that provides features like Unicode support, portability and a cross-platform graphical user interface. All functionality provided by the C++ API is also available from Tcl, thus easing the process of writing applications that access or modify the lexicon. Additionally, the fact that the lexicon is accessible from Tcl enables the incorporation of the lexicon in various Tcl-based text engineering platforms like *Ellogon* [8] or GATE [3]. An application is illustrated in figure 5, where a tool for querying a word form in the lexicon is presented. The user is also able to browse among all morphological entries associated with a specific word form and examine or modify the contained morphological information.

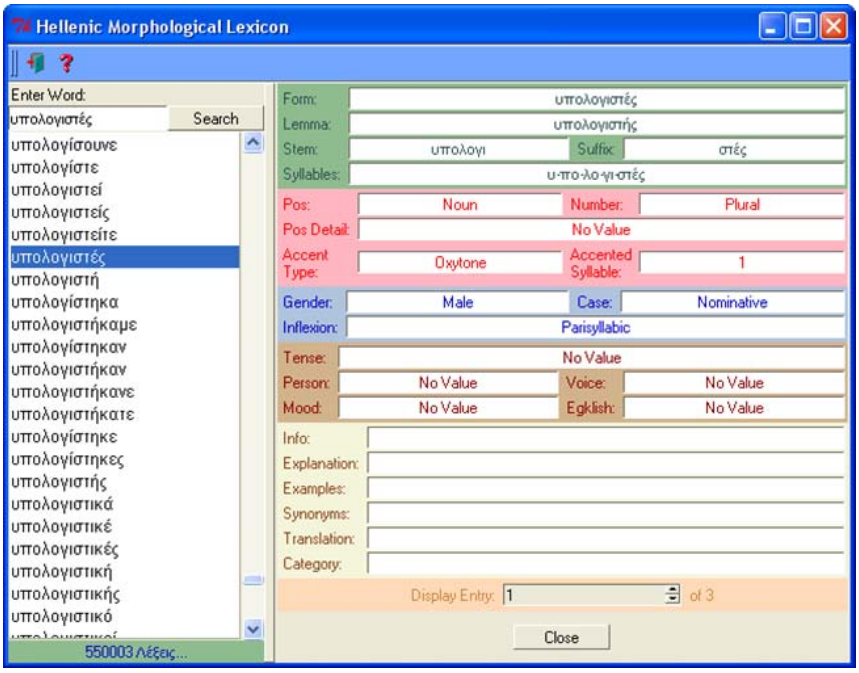

Fig. 5. A tool for querying the lexicon.

2.4 Lemmatisation and Morphological Analysis

The lexicon infrastructure forms a strong basis upon which various task-oriented tools can be easily constructed. In this section we describe two such examples: a lemmatiser and a morphological analyser. Both tools have been developed as components of the *Ellogon* platform but each of them exploits the lexicon in a different way: the lemmatiser extracts and utilises a specialised database from the lexicon, while the morphological analyser accesses the lexicon's database in order to annotate words with all available linguistic information.

The process followed for the creation of the lemmatiser was fairly simple. Initially, a specialised database, that associated word forms with lemmas, was created. A module was also developed that queries all the word forms contained in the lexicon, retrieves the lemma for each word form and fills the specialised database. The lemmatiser requires about 25 MB of memory and it processes ~2.000 words/sec on a PIII/500 PC with 256 MB RAM.

The development of the morphological analyser was also straightforward, as it simply interfaces the lexicon infrastructure with the *Ellogon* platform. The analyser utilises the provided API to query the lexicon about word forms, retrieve the associated morphological entries and pass all the morphological information contained in each entry to the *Ellogon* platform. The component requires about 45 MB of memory, and is able to process ~500 words/sec on the same PC.

3 Lexicon Exploitation by a Greek Controlled Language Checker

A controlled language is a language with a restricted syntax, vocabulary and terminology that is typically applied to technical documents. The aim of using controlled languages in technical documentation is the production of texts with simple structure and restricted vocabulary that can be read and translated more easily [4]. Several software companies (e.g. Bull, IBM) as well as other companies (e.g. Caterpillar, General Motors, Boeing) have been using controlled languages during technical writing of their products documentation. The restrictions imposed by the use of a controlled language preserve uniformity in the writing style, especially in cases where authors tend to follow diverse writing approaches. Additionally, these restrictions reduce ambiguities in the resulting text. The use of a controlled language makes translation faster and of higher quality. A controlled language may also facilitate machine translation, since its resources (vocabulary, terminology and syntax rules) can be exploited by a machine translation system, improving its performance.

In the context of the Greek R&D project SCHEMATOPOIESIS[2], we developed a controlled language checker for the Greek language to assist Greek technical writers as well as to facilitate translation from Greek to other languages [6]. The lexical and grammatical resources of this controlled language cover technical documents from the domain of computational equipment. Technical writers are able to call the checker through their word processor (MS Word is used in the current implementation) as well as through a Web-based application. This allows users to check the format and language of their documents in a similar way as a spelling/syntax checker. The technical document is first converted into an XML format in order to be processed by the checker (Fig. 6). The checker outputs the identified errors in a format "understandable" by the word-processor in order to let users see their errors. The checker checks both text language (correct application of controlled language grammar and vocabulary) and text format (e.g. line spacing, font style and size). The XML text is first processed using linguistic resources (restricted terminology, vocabulary, grammar) and tools (tokeniser, sentence splitter, part of speech tagger, case tagger, morphological analyser, lexical analyser) in order to apply the language checker. Language checking involves lookup of a terminological database (termbase) and a restricted vocabulary as well as checking for paragraph and sentence size, number of sentence clauses, correct appearance of terms, application of syntax restrictions, etc. The text is also checked using a format DTD (Document Type Definition) in order to locate possible errors in format.

At the first stage of the checker's development, we decided to exploit the morphological lexicon as a lemmatiser in order to enrich the output of the part

[2] SCHEMATOPOIESIS is an R&D project funded partially by the Greek General Secretariat of Research & Technology (GSRT) and the EC. The project partners include the Institute for Language & Speech Processing (coordinator), the National Technical University of Athens, NCSR "Demokritos", ALTEC, UNISOFT.

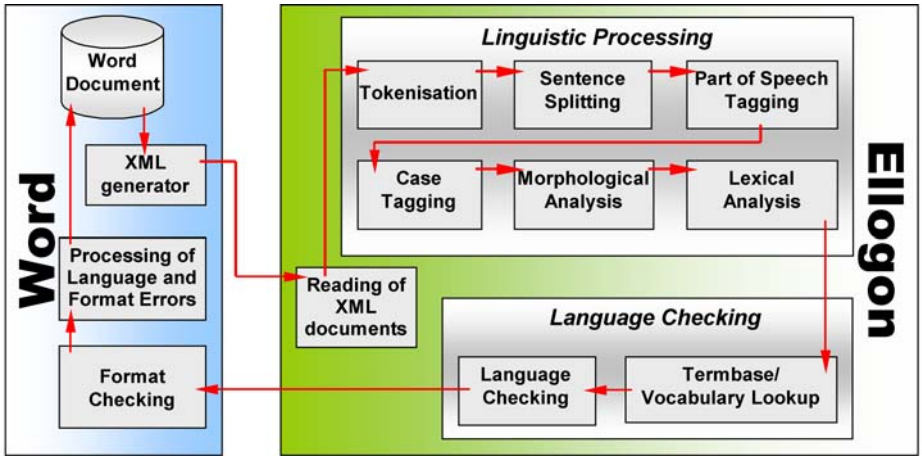

Fig. 6. Architecture of the controlled language checker.

of speech and case tagger with the word lemmas taking into account the lookup module requirements. The lookup module locates those words, phrases or terms that exist in pre-stored lists (in our case the termbase and vocabulary lists). However, in order to reduce the lists size, we maintain only the lemmatised forms of the words appearing there. For instance, there is one entry in the termbase for the term "*τελικός χρήστης*" (end-user) that covers the phrases "*τελικός χρήστης*" (nominative-singular), "*τελικού χρήστη*" (genitive-singular), "*τελικό χρήστη*" (accusative-singular), "*τελικοί χρήστες*" (nominative-plural), "*τελικών χρηστών*" (genitive-plural), "*τελικούς χρήστες*" (accusative-plural). This in turn requires the lemmatisation of the text, since the lookup module attempts to match only the lemmatised forms.

During the evaluation of this version of the checker, we realised that we had to improve the results of linguistic processing in order to improve language checking. This was mainly related to the results of the part of speech and case taggers. Both taggers are based on a machine learning technique, Transformation-Based Error-Driven learning with performance of around 95%. Although this is a good performance for several language engineering tasks (named entity recognition, information extraction), it is not good enough for a task such as controlled language checking. Let's take for instance, one of the rules of the controlled language in the project SCHEMATOPOIESIS that issues an upper limit in the number of consecutive adjectives occurring in a sentence (no more than three). A common mistake of our Greek part of speech tagger concerns the tagging of adjectives as nouns and vice versa due to the morphological similarity of these part of speech categories. Although the tagger is not based only on the morphological form of a word (this is the same for nouns and adjectives) but also on their context, there are several cases where the tagger recognises mistakenly a noun as an adjective. Thus, the technical writer may receive by mistake error

messages concerning the number of consecutive adjectives. However, this affects negatively the general impression that the users have for the checker. This is also the case for the case tagger, which may mistakenly characterise a noun in nominative case although it is in accusative, due to their morphological similarity (the accuracy of the case tagger is ∼93%). Another issue concerns the need to enrich the results of the taggers in order to cover more requirements issued by the controlled language rules. The part of speech tagger is able to identify the following information: part of speech, number and gender for nouns, adjectives and pronouns as well as the tense for verbs. However, the controlled language issues rules concerning the voice and person for verbs, two features that cannot be handled by the part of speech tagger.

The problems mentioned above motivated us to exploit more features of the morphological lexicon apart from the lemma. We had to improve the accuracy of the part of speech and case taggers as well as to enrich their results with more features, such as voice and person for verbs. For this purpose, we developed a morphological analyser as well as a lexical analyser (see Fig. 6). The morphological analyser extracts from the lexicon the required morphological features for those words in the text for which a lexicon entry exists. The lexical analyser, on the other hand, combines the results of both taggers with the results of the morphological analyser. For those words that cannot be analysed by the morphological analyser, we keep the results of the taggers. Concerning those words for which the morphological analyser provides more than one results (e.g. three morphological entries for a noun form that differ in the case: nominative, accusative, vocative) the lexical analyser checks if the tagger agrees with one of these results. If it does agree, this result is kept, otherwise some heuristics are used to select one of the morphological analyser results.

Tokens	Symbols, punctuation marks, digits			2.505 – 15,7%	15990
	Words	Foreign		359 – 2,2%	
		Greek	Analysed	11.351 – 86,5%	
				13.126 – 82,1%	
			Not Analysed	1.775 – 13,5%	

Table 4. Lexicon coverage evaluation.

We evaluated the lexicon coverage as well as the lexical analysis. The lexicon coverage in a corpus of 15.990 tokens is shown in table 4. From these tokens, 15,7% corresponds to symbols, punctuation marks and digits and 2,2% to foreign words (in total 17,9%). From the remaining tokens (Greek words), 86,5% were analysed with the morphological analyser (there was at least one entry for them

in the morphological lexicon) whereas 13,5% were unknown (no entry for them was found in the lexicon).

Concerning the lexical analyser results, compared to the results obtained for the tagger there was a considerable improvement in part of speech (accuracy 97,8%), reducing errors such as the adjective-noun confusion. However, the results were about the same in case identification (accuracy 92,5%), a fact that shows the difficulty of the task for the Greek language. Concerning those features not covered by the taggers (person and voice for verbs) it must be noted that for those verbs that are not known to the lexicon there is no person and voice information.

4 Lexicon Exploitation by an Information Extraction System

Information Extraction (IE) systems fill in predefined data structures with information they extract from unstructured natural language texts that refer to a particular domain. The main processing tasks of an IE system are the following: *Named Entity, Coreference, Template Element, Template Relations and Scenario Template*. The Named Entity task involves the detection and categorization of proper names into predefined domain-dependent semantic categories (e.g. person, location, date). The Coreference task unifies expressions (e.g. proper names, pronouns, definite noun phrases) that refer to identical entities. For each entity, the Template Element task collects particular types of descriptions from the texts, typically pre- and post-modifiers of proper names, like job titles and company descriptions (e.g. "Newton, *a start-up electronics manufacturer*"). The Template Relations task then identifies particular domain-specific semantic relations between template elements; for example, a PRODUCT_OF relation may show a relation between a product and its manufacturer. Finally, the Scenario Template task builds upon the results of the previous tasks to fill an overall template that describes an entire event. For example, the creation of a new joint venture may be seen as an event, that involves PARTICIPANT relations between the new company and each one of the companies that participate in the joint venture, a PRODUCT_OF relation between the new company and a product it will be producing, etc.

In the context of the European R&D project CROSSMARC[3], the SKEL lexicon has been exploited in the construction of a Hellenic Information Extraction system. CROSSMARC applies state-of-the-art language engineering tools and techniques to achieve commercial strength technology for information extraction from web pages, which is applied for e-retail product comparison. The core components of CROSSMARC technology are the following:

[3] CROSSMARC (IST 2000 – 25366) is a R&D project on cross-lingual information extraction applied in e-retail product comparison, funded partially by the EC. CROSSMARC partners include NCSR "Demokritos" (coordinator), University of Edinburgh (UK), University of Roma Tor Vergata (Italy), Informatique CDC (France), VeltiNet (Greece). http://www.iit.demokritos.gr/skel/crossmarc/.

- A Web page collection component, which involves a mechanism for identifying domain-specific e-retailers Web sites and navigating through these sites in order to identify and collect Web pages that describe relevant products.
- A high-quality Information Extraction component for several languages (this is demonstrated in the project's four languages: English, Greek, French and Italian), which locates product descriptions in the collected web pages and extracts important information from them so as to populate a database with information about vendors' offers. The IE component can be adapted semi-automatically to new domains, reducing drastically programming effort and cost.
- A web-based user interface, which processes the user's query, performs user modelling, accesses the databases and presents product information back to the user.

CROSSMARC's technology is demonstrated and evaluated through a prototype e-retail comparison system, based on multi-agent technology, for two different product types (laptops in e-retailers sites, job adverts on companies' sites).

So far, we have used the morphological lexicon in the 1^{st} processing stage of the Hellenic IE system, the Named Entity Recognition and Classification (NERC) task, applied in the 1^{st} domain of the project, that is laptops descriptions in e-retailers web pages [5]. Among different types of information NERC systems utilise information offered by Gazetteers, i.e. tools that identify known entity names in the text and lexical information such as lemmas or stems, which combined with other types of information are the components of rules for NERC grammars. More specifically, the Hellenic NERC (HNERC) module involves three processing stages (Fig. 7):

- Lexical Pre-processing, which includes tokenisation, sentence splitting, lexical analysis (part of speech tagging and lemmatisation).
- Gazetteer lookup, which involves the recognition of known entity names in the text.
- Application of rules for identification and classification of named entities.

The SKEL morphological lexicon has been used as a lemmatiser in the context of lexical analysis (Fig. 7) in order to enrich the output of the part-of-speech tagger with the lemma. More specifically, the lemmatiser produces the lemmas for Greek words found in the pages leaving non-Greek words intact.

Lemmas are used in the rules of the HNERC grammars along with other information from lexical pre-processing, e.g. capitalisation, part-of-speech. The use of a lemma instead of all the different forms of an inflected word reduces the size of the rules. For the same reason of economy, the Gazetteer lookup tool uses the output of the lemmatiser and matches lemmatised lists of known names to the lemmas of the words found in the text.

We recently started the development of the subsequent modules of the IE systems in CROSSMARC, the so-called Fact Extraction (FE) modules. An FE module takes as input the results of the corresponding NERC module (i.e. named entities found in a web pages containing one or more product description) aiming

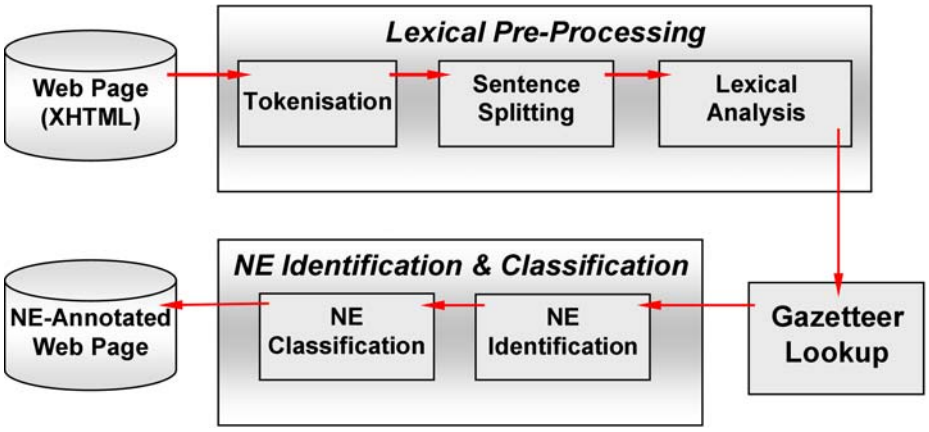

Fig. 7. Architecture of the Hellenic NERC system.

to fill the fields of a pre-defined template with the corresponding information found in a product description (e.g. that a numeric expression of MONEY type identified by the NERC module is the price of the specific laptop offer). We are going to exploit the SKEL lexicon in the hand-written grammars of the Hellenic FE (HFE) module in order to reduce the size of the rules (as we did in the HNERC grammar case). We will also exploit the lexicon in the training of the machine-learning based HFE, using the lemmas of the Greek words occurring in the web pages instead of the exact word forms.

5 Lexicon Exploitation by an Information Filtering System

Information Filtering (IF) systems are used for managing large information flows, presenting users information related to their interests. Many IF systems have been developed in recent years for various applications: news filtering, e-mail messages filtering, web pages filtering, etc. We used information filtering techniques in the context of the R&D project ADIET[4].

The objective of ADIET was the development of a decision support system to analyse share-holdings between companies, exploiting the relevant expertise in the area of the French partner Informatique-CDC. This system visualises the companies' stock connections as a network, the nodes of which are the companies and the links between them represent the percentage of equity capital owned. The ADIET system consists of:

[4] ADIET is a bilateral (Greek-French) R&D project funded partially by the Greek General Secretariat of Research & Technology (GSRT) and the French government. ADIET partners include NCSR, Informatique CDC and Kapa-TEL.

- a text categorisation module for detecting interesting news from the Greek Stock market announcements (Kapa-TEL provides these announcements to their clients),
- a named entity recognition module for detecting the entities involved in the interesting news, and
- a Web-based interface that enables the user to update the network of Greek companies stocks connections, taking into account the categorised announcements and the names of companies identified in them.

In ADIET we exploited information filtering techniques in the text categorisation module in order to identify interesting stock-market announcements. The information filtering system that we developed was based on a machine learning technique that we used originally for the filtering of unwanted e-mails (spam e-mails) [2].

More specifically, a Nave Bayes learner was developed which was trained on a training corpus of manually classified documents. For each document of the training corpus, a vector representation of the form $\overrightarrow{x} = \langle x_1, x_2, x_3, \ldots, x_n \rangle$ was computed, where $x_1, \ldots, x_n$ are the values of the features $X_1, \ldots, X_n$. All features are binary: $X_i = 1$ if feature X_i is present in the document; otherwise $X_i = 0$. Features correspond to words, pairs or triplets of words (1-grams, 2-grams, 3-grams). Therefore, each feature indicates whether or not a certain word (e.g. "$\sigma\upsilon\gamma\chi\acute{\omega}\nu\varepsilon\upsilon\sigma\eta$", the Greek word for "merge") or a sequence of 2 or 3 words (e.g. "$\mu\varepsilon\tau o\chi\iota\kappa\acute{o}\ \kappa\varepsilon\phi\acute{\alpha}\lambda\alpha\iota o$" which means "capital stock", or "$\alpha\acute{\upsilon}\xi\eta\sigma\eta\ \mu\varepsilon\tau o\chi\iota\kappa o\acute{\upsilon}\ \kappa\varepsilon\phi\alpha\lambda\alpha\acute{\iota}o\upsilon$" which means "capital stock growth") occurs in the current document. In order to select the appropriate features (words or sequences of words) for each text category, we calculated the Information Gain (IG) of every candidate feature. For each event type, the features with the highest IG scores were selected.

During training, each Greek word in the training corpus was replaced by its lemma. Consider for example that the four different forms of the lemma "$\kappa\varepsilon\phi\acute{\alpha}\lambda\alpha\iota o$" ($\kappa\varepsilon\phi\acute{\alpha}\lambda\alpha\iota o$, $\kappa\varepsilon\phi\alpha\lambda\alpha\acute{\iota}o\upsilon$, $\kappa\varepsilon\phi\acute{\alpha}\lambda\alpha\iota\alpha$, $\kappa\varepsilon\phi\alpha\lambda\alpha\acute{\iota}\omega\nu$) occur once within a document. During training, each of the four word forms will be treated as the same word (the lemma), increasing in turn the IG for the corresponding feature.

A separate Boolean classifier was constructed for each text category, using the features that were selected for that category. Incoming documents are first processed by a tokeniser and a lemmatiser and then they are classified into one or more categories, based on the results of the corresponding classifiers.

We conducted two sets of experiments. In the first one, features corresponded to single words, while in the second one features correspond to words, pairs of words and triplets of consecutive words. There was no significant difference in the results, which indicates that adding features for two or three consecutive words has no significant effect. Results were very good for all the text categories (F-measure $> 90\%$).

6 Conclusions and Future Work

In this paper, we presented the main characteristics of the SKEL morphological lexicon and described its exploitation by three different natural language applications (controlled language checking, information extraction, information filtering).

Efficient access to the lexicon was one of our main objectives in order to facilitate its exploitation. The integration of the lexicon in the *Ellogon* text engineering platform has facilitated the development of the necessary tools that use the content of the morphological lexicon (lemmatiser, morphological analyser). The efficient update of the lexicon was another issue we focused on. For this reason, we developed a user-friendly interface for adding new lexicon entries.

During the first stages of the lexicon development, we focused on nouns and adjectives since our objective was to improve the accuracy of the lookup modules we used in text processing applications. For instance, in the controlled language checker the lookup module uses lists of terms that are mainly comprised of nouns and adjectives. This is also the case in the gazetteer lookup module and the grammar of the named entity recogniser in the CROSSMARC information extraction system, as well as in the classifiers of the ADIET information filtering system. We plan to update the lexicon with the addition of new entries for verbs. We will also improve the lexicon structure concerning verb entries since in its current state it cannot handle all verb types.

In the context of the CROSSMARC project, we will exploit the lexicon in the development of the Hellenic Fact Extraction module for both domains examined (laptops and job offers).

Finally, we are also examining the use of the lexicon by a natural language generator for Greek. In the context of the M-PIRO[5] project we are developing natural language generation technology that allows personalized descriptions of museum exhibits to be generated in several languages (English, Greek and Italian in the current implementation), starting from symbolic, language-independent information stored in a database, and small fragments of text. In the context of M-PIRO we have developed a Greek lexicon for morphological generation according to the formalism imposed for all languages by the ILEX natural language generation system. We would like to examine the merging of the 2 lexicons, the SKEL lexicon used so far for morphological analysis and the M-PIRO Greek lexicon used for morphological generation in order to have a common resource for both types of morphological processing.

[5] M-PIRO (Multilingual Personalised Information Objects) is a project of the Information Societies Programme of the European Union, running from February 2000 to January 2003. The project's consortium consists of the University of Edinburgh (UK, coordinator), ITC-irst (Italy), NCSR "Demokritos" (Greece), the University of Athens (Greece), the Foundation of the Hellenic World (Greece), and System Simulation Ltd (UK).

References

[1] Anagnostopoulou D., Desipri E., Labropoulou P., Mantzari E. and Gavrilidou M.: "LEXIS–Lexicographical Infrastructure: Systematizing the Data", *Proceedings Workshop on Computational Lexicography and Multimedia Dictionaries (COMLEX 2000)*, pp.63-66, Greece, September 2000.

[2] Androutsopoulos I., Paliouras G., Karkaletsis V., Sakkis G., Spyropoulos C.D. and Stamatopoulos P.: "Learning to Filter Spam E-Mail: a Comparison of a Naive Bayesian and a Memory-Based Approach", *Proceedings Workshop Machine Learning and Textual Information Access, European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD)*, pp.1-13, Lyon, France, 2000.

[3] Cunningham H., Humphreys K., Gaizauskas R. and Wilks Y.: "GATE - a TIPSTER-based General Architecture for Text Engineering", *Proceedings TIPSTER Text Program (Phase III) 6 Month Workshop*, DARPA, Morgan Kaufmann, CA, 1997.

[4] Eijk P.: "Controlled Languages in Technical Documentation", *Elsnews, the Newsletter of the European Network in Language and Speech*, pp.4-5, February 1998.

[5] Farmakiotou D., Karkaletsis V., Samaritakis G., Petasis G. and Spyropoulos C.D.: "Named Entity Recognition from Greek Web Pages", *Proceedings 2nd Hellenic Conference on AI (SETN-02)*, Companion Volume, pp.91-102, Thessaloniki, Greece, April 2002.

[6] Markantonatou S., Karkaletsis V. and Maistros Y.: "An Authoring Tool for Controlled Modern Greek", *Proceedings 2nd Hellenic Conference on AI (SETN-02)*, Companion Volume, pp.165-176, Thessaloniki, Greece, April 2002.

[7] Ntoulas A., Stamou S., Tsakou I., Tsalidis Ch., Tzagarakis M. and Vagelatos A.: "Use of a Morphosyntactic Lexicon as the Basis for the Implementation of the Greek Wordnet", *Proceedings 2nd Conference on Natural Language Processing (NLP 2000)*, pp.49-58, Patras, Greece, 2000.

[8] Petasis G., Karkaletsis V., Paliouras G., Androutsopoulos I. and Spyropoulos C.D.: "Ellogon: a Text Engineering Platform", *Proceedings 3rd Language Resources and Evaluation Conference (LREC 2002)*, pp.72-78, Las Palmas, Spain, May 2002.

[9] Sgarbas K., Fakotakis N. and Kokkinakis G.: "A Straightforward Approach to Morphological Analysis and Synthesis", *Proceedings Workshop on Computational Lexicography and Multimedia Dictionaries (COMLEX 2000)*, pp.31-34, Greece, September 2000.