

In Y. Manolopoulos and S. Evripidou (Eds.), *Proceedings of the 8th Panhellenic Conference in Informatics*, Nicosia, Cyprus, vol. 1, pp. 80–89, 2001.

A GREEK MORPHOLOGICAL LEXICON AND ITS EXPLOITATION BY A GREEK CONTROLLED LANGUAGE CHECKER

George Petasis, Vangelis Karkaletsis, Dimitra Farmakiotou,
George Samaritakis, Ion Androutsopoulos and Constantine D. Spyropoulos

*Software and Knowledge Engineering Laboratory
Institute of Informatics and Telecommunications
National Centre for Scientific Research "Demokritos"
GR-153 10 Aghia Paraskevi, Athens, Greece*

e-mail: {petasis, vangelis, dfarmak, samarita, ionandr, costass}@iit.demokritos.gr

Abstract:

This paper presents a large-scale Greek morphological lexicon, developed by the Software & Knowledge Engineering Laboratory (SKEL) of NCSR "Demokritos". The paper describes the lexicon architecture, the procedure followed to develop it, as well as the provided functionalities to update it. The morphological lexicon was used to develop a lemmatiser and a morphological analyser that were included in a controlled language checker for Greek. The paper discusses the current coverage of the lexicon, as well as remaining issues and how we plan to address them. Our long-term goal is to produce a wide-coverage morphological lexicon of Greek that can be easily exploited in several natural language processing applications.

Keywords: Lexicon, Natural language processing, Morphology, Controlled language

Designated track: Research Track

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 years, Modern Greek lexicons

have been developed by the Computer Technology Institute, the Institute for Language & Speech Processing, the Wire Communication Laboratory – University of Patras, the Software & Knowledge Engineering Laboratory - NCSR "Demokritos". The development of computational lexicons (i.e. lexicons that can be exploited by language processing applications) is a difficult task and becomes even more difficult due to the characteristics of Modern Greek morphology. The complex inflectional system, the existence of marked stress, the free-word-order, the use of old (from Ancient Greek) and new word forms are some of the main characteristics of Modern Greek morphology.

The lexicon of the Computer Technology Institute (CTI) contains ~80.000 lemmas (~1.000.000 word-forms) (Ntoulas et al. 2000). Given a word-form, the CTI lexicon returns the corresponding lemma (or lemmas in case of lexical ambiguity) along with morphosyntactic information, i.e. part of speech, number, gender, case, person, tense, voice, mood, etc. The CTI lexicon was used as the basis for the Greek spelling checker adopted by Microsoft for its word-processor MS Word. This lexicon is also currently used 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) was developed in the context of the EC project LE-PAROLE, aiming to be exploited in natural language processing applications. It contains ~ 20.000 lemmas encoded at the morphological and the syntactic level according to the PAROLE/SIMPLE model (Gavrilidou et al. 1998). The ILSP lexicon is currently being extended in the context of the EPET-II project LEXIS. The new version of the lexicon, by the end of the LEXIS project, will be comprised of ~60.000 entries containing morphological information of which a subset will also contain syntactic information and semantic information (Anagnostopoulou et al. 2000).

The lexicon of the Wire Communication 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 (DWAG). This lexicon was exploited in the context of the EPET-II project MITOS (<http://www.iit.demokritos.gr/skel/mitos/>) for the development of a fast morphological analyser (Sgarbas et al. 2000). The morphological analyser results are used by the MITOS information extraction system (see MITOS demonstrator at <http://www.toureco.gr/Tests/>).

In this paper, we present the lexicon developed by the Software & Knowledge Engineering Laboratory (SKEL) of NCSR "Demokritos" (see SKEL web site at <http://www.iit.demokritos.gr/skel/>). The SKEL lexicon consists of ~60.000 lemmas that correspond to ~710.000 different word forms. It must be noted that the lexicon development was done in parallel with the development of a general-purpose text engineering platform named *Ellogon* (Petasis et al. 2001) which facilitates the development of new tools as well as the integration of these tools in different applications. The SKEL lexicon or parts of it can be easily embedded in different applications taking advantage of the facilities provided by the *Ellogon* text engineering platform.

The SKEL lexicon architecture, the procedure followed to develop it, as well as the provided functionalities to update it, are presented in Section 2. The morphological lexicon was used to develop a lemmatiser and a morphological analyser, which were integrated in a controlled language checker for Greek. Section 3 discusses the lexicon exploitation by the controlled language checker, and the current lexicon coverage. Finally Section 4 presents remaining issues and how we plan to address them. Our long-term goal is to produce a wide-coverage morphological lexicon of Greek that can be easily exploited in several natural language processing applications.

2 SKEL Lexicon for the Greek language

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 (see Figure 1). It is organised around a morphological database, which associates word forms with sets of morphological entries. The morphological database comprises of a fixed number of *pages*. As each page is associated with a unique word form, their number is exactly the same as the number of different word forms the lexicon can recognise. Every page contains a set of morphological entries (see Figure 2). Each morphological entry contains a fixed number of fields, where each field represents a morphological feature. A complete list of available fields as well as all their corresponding values is presented in table 1. For example, the page presented in figure 2 describes all possible features of the word form “αβαθής”: all three entries have the same values for all the features (i.e. the same lemma, part-of-speech, number, etc.), except from the case feature, as the same word form can appear in texts having three different case values. The number of entries in each page is the same as the number of all possible word form instantiations the lexicon is aware of.

During a word form search, the query component tries to locate the page that describes the word form requested. If such a page is located (the word form is contained in the morphological database), all its morphological entries are returned.

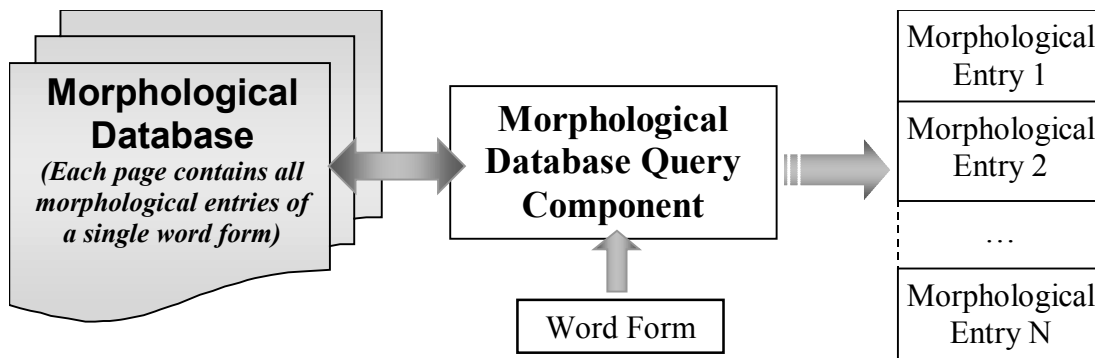


Figure 1: The lexicon Query Component.

<i>Word Form: αβαθής</i>		
Morphological Entry 1		
Part of Speech =	Number =	Gender =
POS_ADJECTIVE	NUMBER_SINGULAR	GENDER_NEUTRAL
Case =	Lemma =	...
CASE_ACCUSATIVE	αβαθής	
Morphological Entry 2		
Part of Speech =	Number =	Gender =
POS_ADJECTIVE	NUMBER_SINGULAR	GENDER_NEUTRAL
Case =	Lemma =	...
CASE_NOMINATIVE	αβαθής	
Morphological Entry 3		
Part of Speech =	Number =	Gender =
POS_ADJECTIVE	NUMBER_SINGULAR	GENDER_NEUTRAL
Case =	Lemma =	...
CASE_VOCATIVE	αβαθής	

Figure 2: A page from the morphological database, describing the word form “αβαθής”.

Morphological Entry Field	Available Field Values
Word Form	The word form
Lemma	The word lemma
Stem	The word stem
Suffix	The word suffix
Syllabication	The word syllabication
Part of Speech	POS_ARTICLE, POS_NOUN, POS_ADJECTIVE, POS_PRONOUN, POS_VERB, POS_PARTICIPLE, POS_ADVERB, POS_PREPOSITION, POS_CONJUNCTION, POS_PARTICLE
Part of Speech Detail	PRONOUN_PERSONAL, PRONOUN_POSSESSIVE, PRONOUN_RE, PRONOUN_DEFINITE, PRONOUN_I, PRONOUN_W, PRONOUN_QUESTIONAL, PRONOUN_INDEFINITE
Number	NUMBER_SINGULAR, NUMBER_PLURAL
Inflectional Type	ACCENT_OXYTONO, ACCENT_PAROXYTONO, ACCENT_PROPAROXYTONO
Accented Syllable	The accented syllable number, counting from the end of the word form
Gender	GENDER_MALE, GENDER_FEMALE, GENDER_NEUTRAL
Case	CASE_ACCUSATIVE, CASE_GENITIVE, CASE_DOTIKH, CASE_NOMINATIVE, CASE_VOCATIVE
Inflection	INFLECTION_EQSYL, INFLECTION_NEQSYL, INFLECTION_IRREGULAR, INFLECTION_ARXAIOKLITO, INFLECTION_IDIOKLITO, INFLECTION_EQSYL_NEQSYL, INFLECTION_DIKATALIKTO, INFLECTION_TRIKATALIKTO, INFLECTION_DI_TRIKATALIKTO
Tense	TENSE_PRESENT, TENSE_PAST_CONTINUOUS, TENSE_FUTURE_CONTINUOUS, TENSE_FUTURE, TENSE_PAST, TENSE_PRESENT_PERFECT, TENSE_PAST_PERFECT, TENSE_FUTURE_PERFECT
Person	PERSON_FIRST, PERSON_SECOND, PERSON_THIRD
Voice	VOICE_ACTIVE, VOICE_PASSIVE
Mood	MOOD_ACTIVE, MOOD_PASSIVE, MOOD_MIDDLE, MOOD_NEUTRAL
Egklish	EGKLISH_ORISTIKH, EGKLISH_YPOTAKTIKH, EGKLISH_PROSTAKTIKH, EGKLISH_METOXH, EGKLISH_APAREMPHATO
Info	Various messages regarding morphological categories
Translation	An English translation, if available
Explanation	Currently unused
Examples	Currently unused
Synonyms	Currently unused

Morphological Entry

Table 1: Morphological Entry fields and their permissible values.

The generation lexicon component is responsible for the generation of all the possible word forms for a given word lemma. Except from the lemma, this component also requires the classification of the given lemma in one of the predefined morphological categories contained in the morphological categories database (see Figure 3). Each morphological category contains instructions describing how the various word forms can be generated from the word lemma and what morphological features 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”. Having 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 “αβαθής” the word form “αβαθές” would be generated three times but the accompanying morphological entries would have been the ones presented in figure 2.

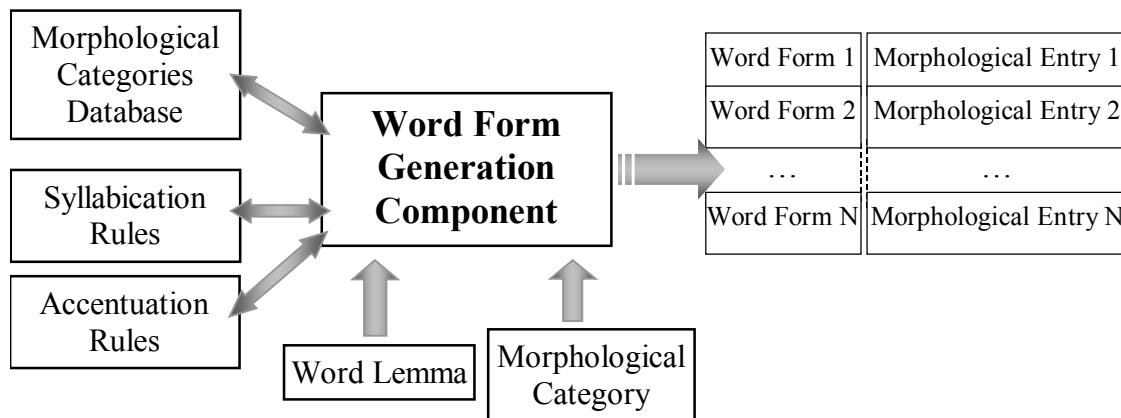


Figure 3: The lexicon Generation Component.

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.

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 were 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 contained 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 by our laboratory. Currently, the list of word lemmas contains approximately 60.000 unique word lemmas.

After the list of word lemmas had been created, 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. 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 morphological categories are inadequate, the user can create a new category and classify the word lemma in this category.

The last step of this process was to give the morphologically classified word lemmas to 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).

2.3 Lexicon Access & Maintenance

Both the query and the generation components as well as the whole software infrastructure of the lexicon has been developed in the C++ programming language, as our main concern was to build a portable and efficient system that will also be able to 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, a lemmatiser can be extracted from the lexicon that only 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* bindings). *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 NLP platforms like GATE (Cunningham 1997) or Ellogon (Petasis et al. 2001). Examples of applications that access or modify the lexicon are illustrated in figures 4 and 5. In Figure 4 a tool for querying a word form in the lexicon is presented, where 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. In Figure 5 a specialised viewer is presented that displays the output of a morphological analyser component based on the lexicon, for the Ellogon platform.

2.4 Lemmatisation, Morphological Analysis

The lexicon infrastructure, as well as the *Tcl* bindings, can operate as a strong basis upon which various task-oriented tools can be easily constructed. In this section we describe two examples of such exploitation, a lemmatiser and a morphological analyser. Both tools are developed as components of the Ellogon platform but each one exhibits a different use of the lexicon: the lemmatiser extracts and utilises a specialised database from the lexicon, containing only a small fragment of the overall lexicon information, 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. This database was written in C and was also embeddable in *Tcl*. The code is quite simple (approx. 200 commented lines) as it utilises infrastructure offered by *Tcl*. Next, a small *Tcl* script was written. This script queries all the word forms contained in the lexicon, retrieves the lemma for each word form and fills the specialised database of the lemmatiser. The performance of the extracted lemmatiser is

adequate: it requires about 25 MB of memory and it is able to process approx. 2,000 words per second on a PC having as CPU a PIII/500 MHz and 256 MB of RAM memory.

On the other hand, the development of the morphological analyser was 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 is coded in C++ and the required code has a length of about 700 documented lines of code, although the largest part of the code is specific to the Ellogon platform. The performance of this component is not as good as the performance of the lemmatiser since it requires about 45 MB of memory, and is able to process about 500 words per second on the same PC.

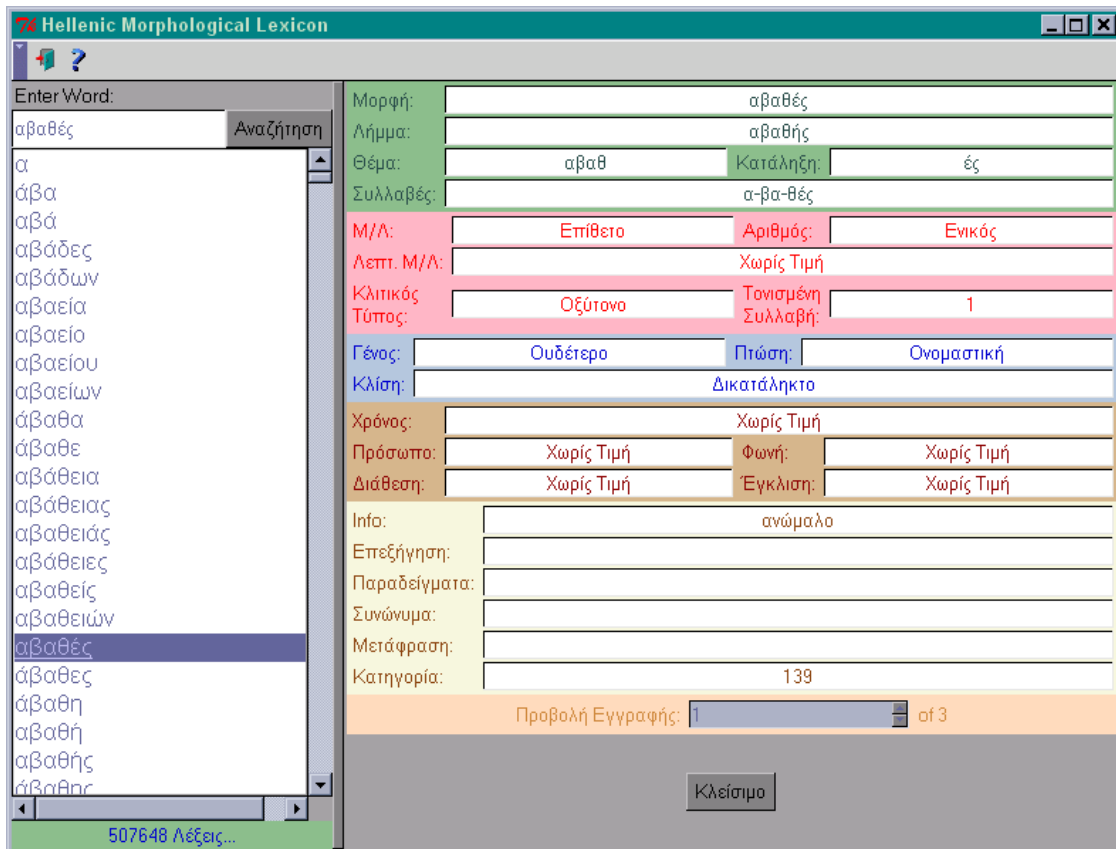


Figure 4: A tool for querying the lexicon (from the “Ellogon” text engineering platform).

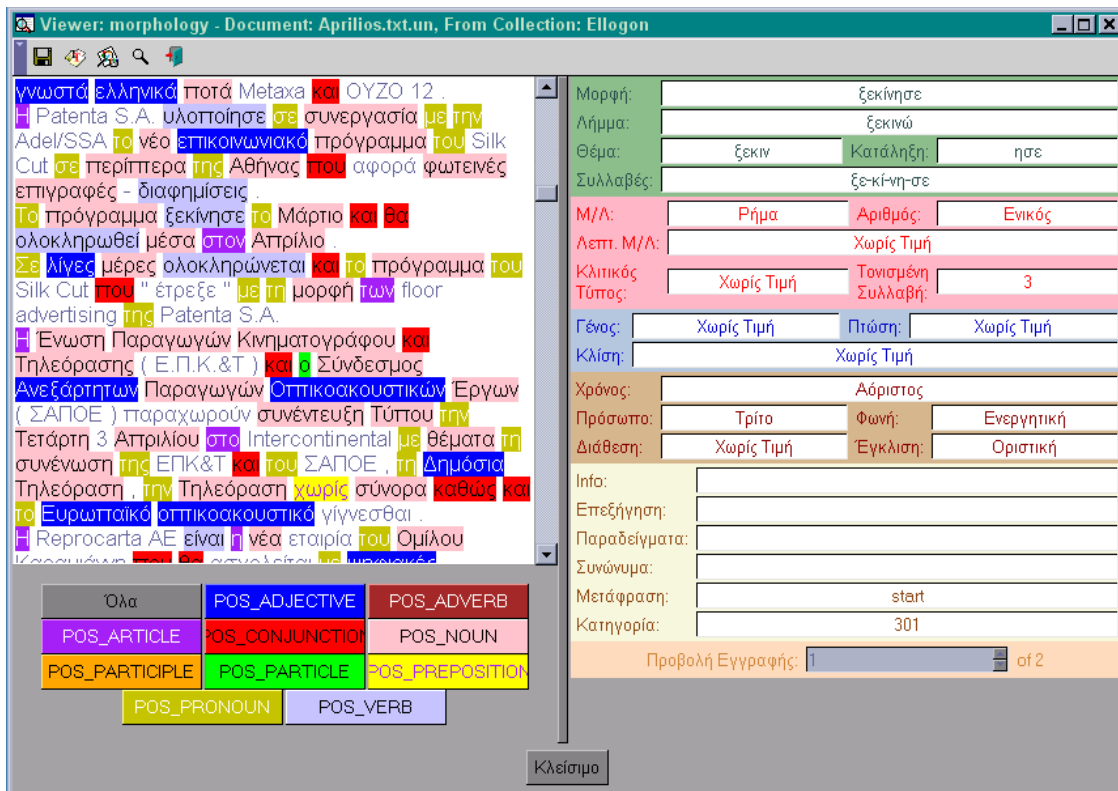


Figure 5: A viewer displaying the results of a morphological analyser based on the lexicon (from the “Ellogon” text engineering platform).

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 (Eijk, 1998; Vouros et al. 1997). 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. The restrictions imposed by the use of a controlled language help to preserve uniformity in the writing style, especially in cases where authors tend to follow diverse writing approaches, and to reduce ambiguities in the resulting text. The use of a controlled language makes translation faster and of a higher quality. A controlled language can also facilitate machine translation systems since the resources provided for it (vocabulary, terminology and syntax rules) can be embedded into the machine translation system, improving its performance.

In the context of the Greek R&D project SCHEMATOPOIESIS¹, 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. Its lexical and grammatical resources cover technical documents from the domain of computational equipment. Technical writers are able to call the checker through their word processor (Microsoft Word is used in the current implementation).

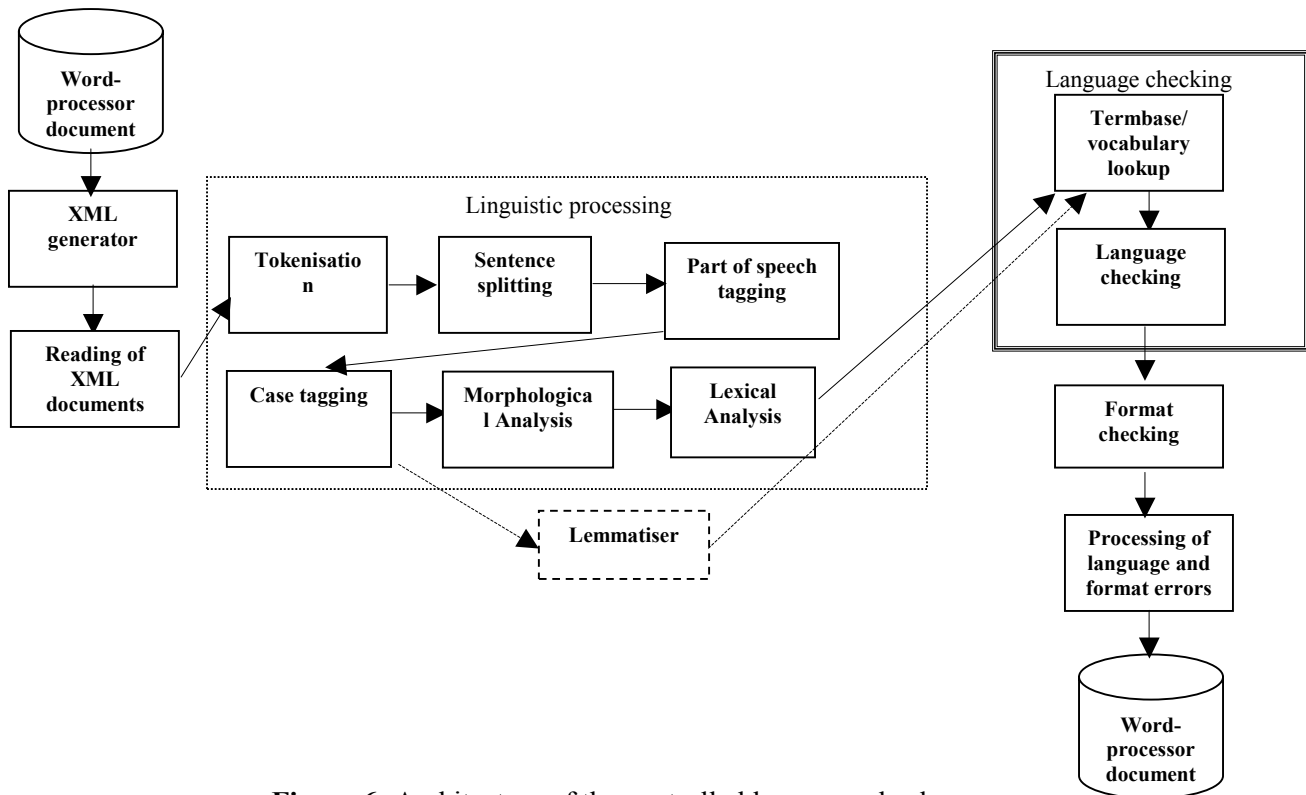


Figure 6: Architecture of the controlled language checker

This allows the user to check the format and language of his/her 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 (see figure 6). The checker outputs the identified errors in a format “understandable” by the word-processor in order to let the user see his/her errors (in the current implementation as Microsoft Word comments). The checker checks both text language (correct application of controlled language grammar and vocabulary) and text format (e.g. line spacing, fonts style and size). The XML text is first processed using linguistic resources (restricted terminology, vocabulary, grammar) and tools (tokenizer, sentence splitter, part of speech tagger, case tagger, morphological analyser, lexical analyser) in order to apply the language checker.

Language checking involves lookup to a termbase and to 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 of speech and case tagger with the word lemmas taking into account the lookup module requirements (this is shown in dashed lines in Fig. 5). 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 size of these lists, we maintain only the lemmatized forms of the words appearing there. For instance, there is one

¹ SCHEMATOPOIESIS is an R&D project funded partially by the Greek General Secretariat of Research & Technology (GSRT). The project partners include Institute for Language & Speech Processing (coordinator), National Technical University of Athens, NCSR "Demokritos", ALTEC, UNISOFT.

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 lemmatization of the text, since the look up module attempts to match only the lemmatized forms.

During the evaluation of this version of the controlled language checker, we realized that we had to improve the results of linguistic processing in order to improve the language checking process. 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 (Petasis et al., 1999). This is a technique used successfully in several other languages apart from Greek, such as English (Samuelson & Voutilainen 1997), German (Schneider & Volk 1998), French (Chanod & Tapananainen 1995). We have found the performance of the part of speech tagger to be around 95% (we found similar results in other types of corpora as it is shown in (Petasis et al. 1999)). 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). One of the common mistakes of our Greek part of speech tagger concerns the tagging of adjectives as nouns or 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 recognizes mistakenly a noun as an adjective. Thus, it is possible that the technical writer will receive by mistake error messages concerning the number of consecutive adjectives. However, this affects negatively the general impression that the users have for the controlled language checker. This is also the case for the case tagger, which may mistakenly characterize a noun in nominative case although this is in accusative, due to their morphological similarity (case tagger accuracy ~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 tagger.

The above mentioned problems 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. 5). 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 the 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 interpretations for a noun 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 heuristic rules are used to select one of the morphological analyser results.

We evaluated the lexicon coverage as well the evaluation of the lexical analyser results compared to the taggers results. The results of the lexicon coverage evaluation in a corpus of 15990 tokens are shown in Table 3. 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 from the morphological analyser (there was at least one entry for them in the morphological lexicon) whereas 13,5% were unknown (no entry in the lexicon).

Tokens	Symbols, punctuation marks, digits			2505 15,7%	15990	
	Words	Foreign		359 2,2%		
		Greek	analysed	11351 86,5%		13126 82,1%
			not analysed	1775 13,5%		

Table 3. Lexicon coverage evaluation

Concerning the evaluation of the lexical analyser results, compared to the tagger results there was a considerable improvement concerning part of speech (accuracy 97,8%), reducing errors such as the adjective-noun confusion. However, the results were about the same concerning 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. So the checker won't be able to identify possible relevant errors.

4 Conclusions and Future Work

In this paper we presented the main characteristics of the SKEL lexicon and described its exploitation by a Greek controlled language checker. Efficient access to the lexicon was one of our main objectives in order to facilitate its exploitation by text processing applications. The integration of the lexicon in the Ellogon text engineering platform facilitates the development of new practical applications. The controlled language checker used under a general-purpose word processor is such an example. 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 lookup modules we used in text processing applications. For instance, in the controlled language checker the look up module uses lists of terms that are mainly comprised of nouns and adjectives. This is also the case in the gazetteer lookup module used by a named entity recogniser (Farmakiotou et al. 2000). We will update the lexicon adding new entries for verbs. We will also improve the lexicon structure concerning verb entries since in its current state it cannot handle all the verb types.

An interesting issue for investigation is the combination of the lexicon with the part of speech tagger. This is currently performed by the lexical analyser module that combines the tagger results with the morphological analyser results. The resources used by the current version of the part of speech tagger include a lexicon, a grammar of lexical rules and a grammar of contextual rules. These resources were created during the training phase of the tagger. More specifically, the lexicon contains all the words in the training corpus associated with their most frequent POS tag, as it was measured from the training corpus. Another option would be to use the lexicon in order to feed the part of speech tagger, i.e. to replace the lexicon learned during the training stage with the morphological lexicon. We believe that this will improve the lexical and contextual rules acquired during the training stage, improving in turn the tagger's performance.

References

(Anagnostopoulou et al. 2000) Anagnostopoulou D., Desipri E., Labropoulou P., Mantzari E., and Gavriliidou M., "LEXIS-Lexicographical infrastructure: systematizing the data", Proc.

- COMLEX 2000, Workshop on Computational Lexicography and Multimedia Dictionaries, Kato Achaia, Greece, pp.31-34, September 22-23, 2000.
- (Brill 1995) Brill, E., Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging, *Computational Linguistics* 21, (1995).
- (Chanod & Tapananainen 1995) Chanod J. and Tapanainen P., Tagging French – comparing a statistical and a constraint-based method, *Proceedings of EACL-95*, Dublin (1995).
- (Cunningham 1997) Cunningham H., Humphreys K., Gaizauskas R., Wilks Y. 1997. GATE – a TIPSTER-based General Architecture for Text Engineering. In *Proceedings of the TIPSTER Text Program (Phase III) 6 Month Workshop*. DARPA, Morgan Kaufmann, California.
- (Eijk 1998) Eijk P., 1998. Controlled Languages in Technical Documentation. *Elsnews, the Newsletter of the European Network in Language and Speech*, Feb. 1998, pp. 4-5.
- (Farmakiotou et al. 2000) D. Farmakiotou, V. Karkaletsis, J. Koutsias, G. Sigletos, C.D. Spyropoulos and P. Stamatopoulos, "Rule-based Named Entity Recognition for Greek Financial Texts", *Proceedings of the Workshop on Computational lexicography and Multimedia Dictionaries (COMLEX 2000)*, pp. 75-78, Greece, September 22-23, 2000.
- (Gavrilidou et al. 1998) Gavrilidou, M., P. Labropoulou, E. Mantzari and S. Roussou, "The morphological level of a Computational Lexicon", *Proceedings of the Panhellenic Conference of New Information Technologies, NIT '98*, pp. 298-308, Athens, October 1998.
- (Ntoulas et al. 2000) A. Ntoulas, S. Stamou, I. Tsakou, Ch. Tsalidis, M. Tzagarakis, and A. Vagelatos, "Use of a Morphosyntactic Lexicon as the Basis for the Implementation of the Greek Wordnet", *Proceedings of the 2nd International Conference on Natural Language Processing (NLP 2000)*, Patra, Greece, *Lecture Notes in Artificial Intelligence*, 1835, pp. 49-58, Springer, 2000.
- (Petasis et al. 2001) G. Petasis, V. Karkaletsis, G.Paliouras and C. D. Spyropoulos "Ellogon: A Text Engineering Platform", NCSR "Demokritos" Technical Report, April 2001.
- (Petasis et al. 1999) Petasis G., Paliouras G. Karkaletsis V., Spyropoulos C.D. and Androutopoulos I., "Using Machine Learning Techniques for Part-of-Speech Tagging in the Greek Language", *Proceedings of the 7th Hellenic Conference on Informatics*, Ioannina, Greece, 1999.
- (Samuelson & Voutilainen 1997) Samuelsson C. and Voutilainen A., Comparing a linguistic and a stochastic tagger, *Proceedings of ACL/EACL Joint Conference*, Madrid (1997), 246-253.
- (Schneider & Volk 1998) Schneider G. and Volk M., Adding Manual Constraints and Lexical Look-up to a Brill-Tagger for German, *Proceedings of the ESSLLI-98 Workshop on Recent Advances in Corpus Annotation*, Saarbrücken (1998).
- (Sgarbas et al., 2000) Sgarbas K., Fakotakis N. and Kokkinakis G., "A Straightforward Approach to Morphological Analysis and Synthesis", *Proc. COMLEX 2000, Workshop on Computational Lexicography and Multimedia Dictionaries*, Kato Achaia, Greece, pp.31-34, September 22-23, 2000.
- (Vouros et al. 1997) Vouros G., Karkaletsis V., and Spyropoulos C.D., 1997. Documentation and Translation. *Software without frontiers*, P.A.Hall and R.Hudson (eds), J.Wileys & Sons, 1997, pp. 167-202.