Abstract

This paper presents the design and implementation of terminological and specialized textual resources that were produced in the framework of the Greek research project “IATROLEXI”. The aim of the project was to create the critical infrastructure for the Greek language, i.e. linguistic resources and tools, to be used in high level Natural Language Processing (NLP) applications in the domain of Biomedicine. The project was built upon existing resources that were developed by the project partners (and further enhanced within its framework), i.e. a Greek morphological lexicon of about 100,000 words, and language processing tools such as a lemmatiser and a morphosyntactic tagger, and it additionally developed new assets such as a specialized corpus of biomedical texts and an ontology of medical terminology.

Keywords

Biomedical terminology, data mining, knowledge representation, ontologies.

1. Introduction

The quantity of biomedical data which is daily produced by the medical society, i.e. health institutions, educational organisms and research institutes, has been enormously enlarged especially during the last decade. This data which is mainly available in digital form and mostly accessible through Internet in nowadays has been characterized by Eysenbach [1] as “information jungle” of narrative form, due to its enormous size and (in most
of the occasions) its unstructured form. However, information is only valuable to the extent that it is accessible, easily retrieved and relevant to the users’ interests. The growing volume of data, the lack of structured information, and the information diversity have made information and knowledge management a real challenge towards the effort to support the medical society. It has been realized that added value is not gained merely through larger quantities of data, but through structuring of the data into knowledge for more sophisticated access to the required information.

In particular, knowledge management, data mining as well as text mining techniques have been adopted successfully in recent years leading to the creation of a number of resources and tools. The lack of similar high level language tools is greater in less-used languages like Greek, due to the limited research funding and the restricted interest by the medical industry, and also due to the intrinsic particularities and grammatical characteristics of the Greek language (e.g. highly inflective language, complex morphology, free word order, etc.).

The project IATROLEXI¹ (http://www.iatrolexi.gr) aims at the creation of the critical infrastructure for the Greek language which will constitute the groundwork for advanced NLP applications in the domain of biomedicine, i.e. text indexing, information extraction and retrieval, data mining, question answering systems, etc. To accomplish this, a number of essential tools and resources for the Greek language were constructed, which will allow better management and processing of the digitally encoded information in the biomedical field.

This paper is structured as follows: Section 2 presents some background information on NLP and ontologies in the biomedical domain. Section 3 gives a description of IATROLEXI project’s main outcomes. Finally, in section 4 the conclusions are given.

2. Background

Natural language processing (NLP) has been applied to biomedical text for decades, in fact, soon after computerized clinical record systems were introduced in the mid 1960s [2]. The computerization of clinical records increased the tension in the field of medical reporting and recording. In [3] a broad overview of NLP in
medicine can be found, with emphasis to milestone projects and systems such as the Linguistic String Project, Specialist, Recit, MedLEE, and Menelas. The overview of [4] also concentrates on NLP with emphasis on extraction systems, giving a short summary of earlier projects and the state of the art at that point in time.

Ontologies are considered to be a fundamental prerequisite for advanced language processing, knowledge management and the Semantic Web, since they offer the mechanisms for the formal representation and the description of the concepts in a given domain [5, 6]. Typically, an ontology identifies classes of objects that are important in a domain and organizes these classes in a hierarchy. Each class is characterized by some properties and is related to other classes or to elements of other classes through a number of significant relations. The predominance of ontologies as knowledge sources in information processing lies on their power to represent knowledge in a model that is comprehensible equally by either humans or machines. Thus, ontologies assist communication between human agents, achieving interoperability among computer systems, and advancing the systems' quality performance on indexing, processing, retrieval and extraction of required information.

Several methodologies [7] have been proposed for the engineering of ontologies within a knowledge management setting. From feasibility analysis and identification of goals, to requirements’ specification, implementation, evaluation and maintenance of ontologies, the ontology life-cycle must be clearly defined and further supported by ontology engineering tools [8]. Most recent methodological approaches focus on the application-driven engineering of ontologies, supporting the introduction of ontology-based knowledge management systems.

A significant amount of work in developing an NLP system concerns extending lexical knowledge. Since there is a very large number of words and phrases associated with clinical concepts, the task of adding entries to the lexicon is considerably demanding [9]. The National Library of Medicine in U.S.A. has undertaken a large-scale effort to facilitate access to biomedical information. The development of the UMLS (http://umlsinfo.nlm.nih.gov/) and the release of the SPECIALIST Lexicon will substantially benefit NLP systems. A UMLS concept is given a unique identifier, and all synonymous concepts have the same identifier.

1 "IATROLEXI" project was partially funded by General Secretariat of Research & Development (project code: 9) within Measure 3.3 of "Information Society" Operational Program.

2 Within this framework, the methodology that was adopted for the construction of the ontology and it is described in this paper, was based on a hybrid top-down and bottom-up approach as it is explained in section 3.3.
This feature provides a substantial body of knowledge that NLP systems need: link words in text to a controlled vocabulary (i.e. to the UMLS or to one of the other source vocabularies). The UMLS also has a Semantic Network through which semantic categories are assigned to all concepts. For example, “fever” is assigned the category SIGN/SYMPOTOM. The categorization provides the semantic knowledge needed by NLP systems to identify relevant units of information. The SPECIALIST Lexicon, which has over 250,000 entries, assigns syntactic categories to words and phrases in biomedical text. The lexicon is not only useful for NLP extraction tasks, but also for indexing and vocabulary development.

The UMLS Metathesaurus (part of the UMLS project) is a large database consisting of over 50 medical vocabularies. Among them is Medical Subject Headings (MeSH) vocabulary, which apart from the English part, includes the translations in a number of different languages (e.g. Arabic, Chinese, Dutch, French, German, etc.), in order to make the vocabulary useful for non-English users, or, to put it in another way, to non-English documents [10]. Most of these translations were produced by national medical information centers or equivalent organizations.

3. Project’s main goals: Resources and tools

In order to deal with knowledge representation needs for Greek biomedical texts, it is inevitable that a number of text analysis tools and linguistic resources should have been developed. These tools constitute the basis for any application regarding data mining, NLP, indexing, etc.

3.1 Corpus of biomedical texts

To the best of our knowledge, there were no Greek electronic biomedical corpora structurally or linguistically annotated until the beginning of the project. Thus, within the project's framework, a biomedical corpus was developed, whose main design principles deal with its balance and representativeness, as well as with its annotation.

Balance and representativeness are the main requirements for corpus design. According to these requirements, the scope was to develop a Greek corpus of written texts, coming from all different domains of biomedicine. The corpus should contain documents from as many biomedical text fields as possible. Moreover, recent research makes clear that full-text articles are preferable from abstracts, if we want to build high-recall
text mining systems [11]. Therefore, it seems clear that the IATROLEXI corpus, aimed to biomedical text mining systems, should include full-text articles and not text samples.

Corpus annotation is the distillation procedure that adds (or extracts the) value to the texts. The annotation process of the IATROLEXI corpus involves almost all NLP components either adopted or constructed within the framework of IATROLEXI: a tokenizer, a sentence splitter, a morphosyntactic tagger, a biomedical gazetteer, a multi-word term recognizer, and an ontology-based semantic tagger.

Due to time limitations we considered only documents directly available on Internet sites, thus we recorded portals or other websites that included directories of health-related information. We started our investigation from websites of research and academic institutions, e.g. MedNet Hellas (a Greek Medical Network), Greek National Documentation Center, Library of University of Macedonia, etc. The above sites proved to be very helpful, since they contained a rather exhaustive list of directories of Greek biomedical journals. Next, we utilised popular search engines, such as Google, Yahoo, Live Search, etc. in order to identify additional websites that might contain interesting texts. Through these search engines, we mainly acquired the web addresses of Greek medical conferences that were not listed in the directories mentioned above. Overall, forty websites were identified to contain appropriate medical documents for IATROLEXI corpus. The total number of documents that were collected is touching 6,250, i.e. about 11.5 million words [12] (typically, 500,000 - 5 million words are sufficient enough for a specialized corpus) [13].

The documents collected have been classified as regards to medium and topic. The “medium” classification was more or less straightforward since they were either periodical articles (abstracts or full papers) or conference papers. Regarding the “topic” classification, an appropriate scheme was developed manually by the medical experts, based on medical specialties (cardiology, cytology, neurology, dermatology, paediatrics, etc.). Although the present text collection consists of a large variety of medical topics and subfields, its medium and genre representativeness is still restricted to research (vs. clinical) domain, as it comprises mostly scientific articles from medical journals (70%) and conference abstracts (30%).

3.2 Creation, enhancement and/or adaptation of existing resources and tools

A number of tools have been created, enhanced and/or adapted in order to constitute an environment, supporting: a) the discovery of syntactic patterns that can be candidate multi-word terms (i.e. “Kanon”
formalism for writing rules for multi-word term recognition), b) the construction of the ontology (i.e. adjustments in the Protégé editor, so as to handle Greek characters’ encoding in the ontology (cf. sec. 3.3)), c) the detection of medicine terms in the documents of the corpus (i.e. the development of a concordancer for studying Greek biomedical terms’ collocational patterns), and d) semantic indexing of the documents (i.e. semantic annotator for assigning semantic type information in the terms found in texts). All of the tasks mentioned are actually done in the present work, except for the semantic indexing which is envisaged as a future application (cf. sec. 4). The core mechanism for most of the software components functioning on the corpus is annotation.

The software implementation platform of all NLP components is Java v 1.5. The operational environment integrating and orchestrating the software components working with annotations is the Apache UIMA platform. UIMA stands for Unstructured Information Management Architecture; it was developed by teams from IBM Research and IBM Software Group and is now released to the open-source community as an Apache project (http://incubator.apache.org/uima/).

The main components constructed for the analysis, annotation and indexing of the documents, along with the resources they use, are presented in the following sections. In figure 1, the different steps of the annotation process are presented schematically.

3.2.1 Document conversion

The documents collected from the Internet are either in “html” or in “pdf” format. On the other side, all of the tools process documents in a common format which is pure text decorated with annotations. The UIMA terminology for this common format is CAS (Common Annotation Structure). To satisfy the requirement of feeding the annotation process with documents of a common format, we decided this format to be plain text, for the reason that only the textual content of the documents is of interest; scripting, styling, formatting and page rendering information had to be filtered out. Therefore, we developed two document converters: an “html-to-txt” converter and a “pdf-to-txt” converter. The converters were based on freeware software, though certain modifications were needed in order a) to deal with the various Greek character sets that are currently used and b) to keep as more information as possible from the formatting structure of the documents (e.g. section titles, etc.).
The adoption of plain text format, for our corpus, has several advantages that include among others: less storage needs, easier to process, etc. [14]. However, there are certain disadvantages with most important the loss of any formatting information (e.g. italics, bold, etc.) that might be useful to the analysis of the text. The final decision was to try to keep the tools as simple and less complicated as possible, even if we had to drop some formatting information.

3.2.2 Tokenization and sentence splitting

Content analysis starts with tokenization, i.e. conversion of any character stream to a token stream. Tokenization is carried out in two steps. In the first step, a text stream is roughly converted into a token stream based on white space delimiters and some symbol characters (e.g. dot, comma, question mark, etc.). At the same time, the token orthography is recorded. By “token orthography” we mean the character set (Greek, Latin, or mixed) and the case (upper, lower, all capitals, first capital), e.g. “νόσος” is a Greek-letter-lower-case token, “Disease” is an English-letter-first-capital token, “H.I.V.” is an English-letter-all-capital + middle-dots + ending-dot token. The lack of homogeneity in the input alphabet constitutes a particularity of Greek biomedical texts, where many biomedical multi-word terms include both Greek and Latin tokens (e.g. σύνδρομο Down, βιταμίνη C, πρόδρομοι DNA, etc.). In the second step, the token stream passes through a refinement module. Tokens of a specific orthography may further split into two or three tokens. For example, a token that ends with a comma or question mark or exclamation mark or colon or semi-colon will split into two tokens; a token that starts with a quote and ends with a quote will split into three tokens.

Sentence splitting examines the token stream produced from the second step of tokenization and locates tokens that traditionally play the role of sentence delimiters, i.e. full stops, question marks, exclamation marks and dot-ending tokens. Special care is taken for tokens that end with a dot, so as to decide whether this dot is part of the token (e.g. the token is an abbreviation) or the dot is a punctuation mark (i.e. a full stop). Among the various tests performed towards the disambiguation of the ending dot, the one worth-mentioning refers to tokens where all the characters before the dot are Greek letters. If these letters are more than two and constitute a valid Greek word, then the token splits into two tokens: a Greek-word token and a full-stop token. The validity of a Greek word is examined through lookup in Morphological Lexicon.
3.2.3 *Morphosyntactic tagging*

Morphosyntactic tagging is based on a broad-coverage Morphological Lexicon of Greek (~100,000 words, i.e. ~1.200,000 word-forms) that was developed in previous projects [15, 16]. The contents of the lexicon are organised into lemmas. Each lemma contains all the word-forms of a Greek word accompanied by the values of their morphosyntactic attributes. The basic morphosyntactic attribute of a word-form is its part-of-speech. The value of part-of-speech determines other morphosyntactic attributes that characterise a word-form: gender, number and case for nouns, adjectives, articles, pronouns; voice, tense, mood, number and person for verbs. The first word-form of each morphological lemma, the headword, plays the role of lemma representative. As the morphological lexicon is monolingual, morphosyntactic annotations are assigned only to Greek words.

Each Greek-letter token identified during tokenization is assumed to be a Greek word-form. Every word-form is looked-up in the morphological lexicon. The possible outcomes are three: a) the word-form is found in one morphological entry, b) the word-form is found in two or more morphological entries, and c) the word-form is not found. Instead of trying to assign unambiguous morphosyntactic annotations to word-forms within this task, we left this process to the multi-word recognition task (cf. sec. 3.2.5).

3.2.4 *Biomedical word identification*

The next step was to mark words that belong to the biomedical domain. This marking was crucial for the next processing steps, i.e. the multi-word term recognition and the semantic tagging.

Biomedical words are identified by searching a pre-stored list (gazetteer) that currently contains ~52,000 biomedical word-forms (that correspond to ~9,000 biomedical words). The contents of the gazetteer partly come from the existing morphological lexicon and partly were collected through a process of automatic or semi-automatic term’s extraction described in section 3.3. Every single biomedical word may be a biomedical term by itself (which can be certified through look-up in a biomedical dictionary or ontology) or may be part of a multi-word biomedical term.

---

3 It is worth mentioning that many Greek word-forms which occur in the biomedical texts may not be found in any morphological entry, because they constitute older word-forms and, thus, they have not been encoded in the morphological lexicon; for example, the word-form “οστούν” (bone), which occurs in the multi-word term “ιερό οστούν”, will not be found in the Greek morphological entry of “οστό”. 
3.2.5 Multi-word term recognition

The multi-word recognition mechanism is one of the advanced outcomes of the project. It is a feature based grammar formalism, which was named “Kanon” from the Greek word “κανών” (rule), where every rule recognizes a syntactic pattern in the input text. Rules can be applied in a consecutive and aggregative manner. Consecutive means that rules are applied in the same sequence of annotated text spans repeatedly i.e. as far as we can apply rules and the size of the text span’s sequence is decreased, the processing continues. This scheme together with the context free nature of rules simulates classical LR parsing avoiding cycles or endless reduction loops. Aggregative means that a set of rules can be applied after another set of rules. We can have many levels of parsing permitting us to handle many cases of “context sensitive” syntactic cases of natural languages.

The format of the rules resembles that of the context free BNF rules, where every symbol is presented as a set of feature value pairs. The grammar is strongly “typed” in the sense that every feature has a type which specifies the values of its instances in the rules. The syntax of the rules is depicted in the sample grammar of Table 1.

A rules’ file consists of the following sections:

- **options** section (line 1) affects the way that the rules are processed and used. In the sample grammar (Table 1) we set two options. The name of the grammar rules “Terms” (line 2) specifies the name of the annotator that will apply these rules to text spans. The second option named “maxdepth” (line 3) specifies the number of levels that operators like * (Kleene star) and + will be expanded.

- **types** section (line 4) presents new derived types that features can use. Our formalism uses the primitive types of **number** and **text** and the composite types of **set** and **value**. One of the main characteristics of the formalism is its ability to communicate with the implementation Java environment. The members of the set MATTRS are defined in the interface class and imported to the grammar in line 5.

- **features** section (line 6) defines the names and types of the features to be used in the grammar rules. All features defining grammar symbols in the following rules must be treated in this section. There are no untyped features, as it is already mentioned, and the system accomplishes a strong type checking of how
values and types are used in the rules. In our sample we define two features with names ONTO (line 7) and LEXY (line 8), which both are of type object. ONTO objects used to check the agreement of Gender, Case and Number between participants in a rule and LEXY incorporates all the lexicon entries of a word, i.e. part of speech and morphological information of the word.

- **functions** section (line 9) declares Java functions that can appear in expressions, specifying the values of features in the rules. There are different types of functions as:

  1. **Object** functions can appear in the body symbols (predicates) of a rule. There are object (instance) methods (in Java terminology) that can take a list of parameters and return a value assigned in a feature. Function *HasMArrs* (line 10) in the sample grammar takes as input parameter a set of attributes and it returns *true* or *false* depending on the morphological attributes of the examined token.

  2. **Predicate** functions are static methods of a Java class. They can be presented only in the body predicates. Except from the defined parameter list, these functions are enriched with an extra parameter, which is the table of all feature value pairs assigned to the predicate that they appear in. The function *GNC_Agreement* (line 11) checks the agreement of Gender, Number and Case of the neighbor symbols found in the input. The first parameter appearing in its definition is a number denoting the way this agreement must be checked, i.e. combination of Gender, Number and Case.

- **rules** section (lines 12-18) contains the actual grammar rules. Every rule contains a head predicate and one or more body predicates. Head is defined in terms of the body predicates, which means that if a sequence of symbols (text spans) matches the body predicates, then we can reduce these predicates to the one of the body. Rules are independent of each other. Their order does not matter the way they are evaluated. The system can use different heuristics about which rule to choose for reduction in case that multiple rules match an input sequence of symbols. The current applied technique chooses the longest (in terms of size of predicates in the body of a rule) rule. The symbols ‘\’ and ‘/’ specify the left and right context of a reduction. We can have a list of predicates at the left of the ‘\’ symbol denoting the left context of the reduction. The meaning of the left context is that we expect to match all the predicates presented in the left context but we will not use them in the reduction. The same holds for the right context. Only the predicates
presented between the ‘\’ and ‘/’ symbols will be reduced. Parentheses can also be used to group sequence of predicates. A body predicate or group can be right followed by a repeating operator of the ‘*’, ‘+’, \{m,n\}. The meaning of ‘*’ is zero or more instances of the predicate or group existing in the left of the operator must be matched. The ‘+’ operator is interpreted as one or more instances while the expression \{m,n\} means that we expect to match at least m and an most n instances.

The “Kanon” parser takes as input a unification grammar (written according to the already specified formalism, see Table 1) and produces a compiled representation of the rules. The actual application of the rules is performed by an execution engine, which loads the compiled rules at start-up (i.e. the parser is the execution engine plus the parsing model). The execution engine incorporates a prototype unification algorithm for the efficient handling of multi-valued features, which facilitates the treatment of the inherent morphosyntactic ambiguity (for more on unification, see [17]).

The grammar rules are used either to discover candidate multi-word terms according to a certain syntactic pattern (like the one presented in lines 14-18) or to detect multi-word terms in the text in order to index documents with these terms. For the indexing procedure extra process is needed. The candidate multi-word term is looked up in the lexicon of multi-word terms and, if found, its “id” is used to index the document. The lookup in the lexicon does not follow an exact match approach, in order to avoid the codification of multi-word terms’ inflection and thus the construction of big lexicons. Instead, the lexicon contains only the headwords of the lemmas which constitute the multi-word term, along with the grammar rules that express its syntactic pattern. During the indexing phase, we first detect a syntactic pattern as a candidate multi-word term, then we look it up in the lexicon by using the headwords of the constituting lemmas, and finally we output its “id”.

3.3 The development of a biomedical ontology

Following Uschold and King’s [18] three suggested strategies for building an ontology, namely, the top-down approach, in which the most abstract concepts are identified first, the bottom-up approach, in which the most specific concepts are identified first and the middle-out approach, in which the most important concepts are
identified first, in the present framework the adopted methodology was based on a hybrid top-down and bottom-up approach.

The construction of the ontology was accomplished through two main procedures as follows: 1. the determination of an initial top-level taxonomy which would be used for the knowledge representation of the biomedical domain (i.e. selection of the hierarchy and its description tools, such as classes and relations between concepts), and 2. the compilation of the macrostructure of the ontology, namely the selection of the relevant biomedical terms that designate and instantiate the concepts of the hierarchy nodes.

As regards the first step, since the aim of IATROLEXI was to build a generic and application independent infrastructure for the language processing of the Greek biomedical data, the project team opted for the adoption of the UMLS Semantic Network (SN). Adopting UMLS semantic network as an initial top-level ontology, and mapping it into Greek, we gained access to the conceptual information for some thousands of biomedical concepts. The whole number of the SN semantic types and semantic relations were translated into Greek, while both English and Greek versions of the SN were further processed and evaluated.

For the enrichment of the taxonomy’s nodes with Greek biomedical terms two combined procedures were adopted (see figure 2):

1. Processing of a number of dictionaries and glossaries from which a large number of biomedical terms were extracted. More specifically, we have processed 14,000 terms of the Greek MeSH (1996), which they were still included in the American MeSH® (2006) and a basic vocabulary of 5,000 Greek biomedical terms already included in the Morphological Lexicon. Moreover, in these resources a sufficient number of multilingual terms was detected, which was further used as a basis for the construction of a grammar for the automatic recognition and extraction of multi-word terms. Linguists and medical experts reviewed the relevant vocabularies as regards both their linguistic and medical

---

4 It has to be clarified that the UMLS SN offered us a conceptual skeleton upon which we could build our terminological data, while the design and the support of alignment procedures to the UMLS Metathesaurus was not in the scope of our project.

5 The original version of the Greek MeSH (1996) was comprised of 17,800 Greek equivalents to English terms, some thousands of which had been revised and/or excluded from the American MeSH® version of 2006. For that reason, the processing of the Greek MeSH was confined to the subset of 14,000 terms.
appropriateness and correctness, and they ended-up with a list of 11,000 terms that were incorporated into the ontology.

2. Automatic and semi-automatic candidate term’s extraction from the IATROLEXI corpus (cf. sec. 3.1). This subtask was a repeatable procedure which was feeding the core vocabulary extracted from dictionaries with real terminological data extracted from the above mentioned biomedical corpus. The output of this work was a number of terms which, after having been reviewed by the experts, were incorporated into the ontology nodes. The automatic term extraction was based on the one-word and multi-word term identification tools developed in the framework of this project (as already described in sec. 3.2.4 & 3.2.5).

While the former procedure was applied once, the latter was a continuous circle towards the enrichment of the taxonomy nodes with more terms.

The final version of the IATROLEXI ontology comprises 14,000 concepts, which are represented by 16,500 terms. Although the total number of the ontology population is not large enough as compared, for instance, to other vocabularies integrated to UMLS, such as the French (86,999 terms), Portuguese (58,349 terms) or even the Swedish (25,185 terms) translation of MeSH, the Iatrolexi ontology comprises a substantial number of the basic vocabulary in the biomedical domain, which is not confined only to the vocabulary provided by MeSH but also to vocabulary which found in Greek dictionaries and textual resources.

Each entry in the ontology bears the following information:

- The concept id, which is unique. The concept id for each concept in the Greek ontology is identical to the concept id of the UMLS SN for the top-level nodes, to the MeSH id for the concepts of lower depth, while a distinct id was assigned to concepts that are instantiated by Greek terms drawn from other sources.
- one or more English terms representing that concept
- one or more corresponding Greek terms representing the same concept
- a gloss in English
Relations between the ontology concepts that have been encoded are of two types: the \textit{is_a relation} (e.g. \textit{sex chromosome aberrations} is a type of \textit{Disease}) and the \textit{synonymy relation} (i.e. two or more terms used for the representation of the same concept, such as the Greek term \textit{μυκωτικό ανεύρυσμα} is a synonym of the term \textit{μολυσμένο ανεύρυσμα}, both corresponding to the English term infected aneurysm).

The Protégé editor was used for the insertion and the processing of the initial top-level taxonomy and the OWL formalism used for the representation of the ontology. Since the Protégé software does not allow encoding of Greek characters for classes/semantic types and instances/terms, some adaptations were considered crucial for the representation of the Greek part of the ontology, such as the insertion of certain labels and comments for encoding Greek names. Though, the enrichment of the top-level taxonomy with terms was performed in a database environment, since Protégé was proved difficult and awkward enough to handle term encoding issues. Moreover, a plug-in of Protégé was developed, the “IatrolexiTab”, through which users can browse the ontology using the Greek names for semantic types and terms.

The IATROLEXI ontology per se, is accessible through a web browser at the following address:


4. Conclusions

NLP infrastructure is a key element in the development of computer applications in several areas, such as data/text mining, knowledge-based decision support, terminology management, systems interoperability and integration, etc. A significant body of work exists today that reports on experiences with various approaches in crucial areas of research. On the contrary, in the biomedical field and especially for the Greek language, there is not much work implemented.

Project IATROLEXI aimed to cover this certain gap by developing a number of NLP resources as well as application tools for the scientific community. On the one hand, the scientist may use the outcomes of the project in his/her own way towards his/her specific research needs. On the other hand, the average user may

\footnotesize{\textsuperscript{6} Information concerning the size of the above mentioned national translations of MeSH was drawn from http://www.nlm.nih.gov/research/umls/sourcerelasedocs/}
experience much more powerful ways in searching for biomedical (and not only) information (e.g. using specific terms from the ontology, or combination of monolingual or bilingual terms, or relations that link terms to each other, etc.).

Furthermore, there are many possible applications of the created infrastructure in the biomedical field. They can be divided in certain categories like: text mining with all its potential applications [19], data exchange among applications and information integration [20], comparative studies in biology [21], indexing of medical texts (e.g. MeSH) with special emphasis in interlanguage indexing (e.g. English – Greek), etc. Even without further development, the resources and tools that were produced might be useful in various ways: For example, during the ontology population, the medical experts utilized the collected corpus with the use of a concordancer [19] (the concordancer can be accessed through the http://www.iatrolexi.gr/iatrolexi/webtools/concordancer/index.html). Using this tool, each expert had the ability to search and inspect the use of a certain term within its context, and thus, decide where exactly in the ontology should the term be positioned.

Currently, a part of our efforts focuses on the completion of the multi-word term recognizer. More specifically, in section 3.2.5 we presented the extraction of candidate multi-word terms from the corpus, based solely on linguistic knowledge. In order to improve this process we are thinking of exploiting some type of statistical evidence which will help us to compute a term-validity metric (e.g. the C/NC-value metric, see [22]).

Regarding the ontology three applications are envisaged:

1. **Semantic tagging**: Any term found in the dictionary can receive an annotation that encodes its semantic type and thus links the term with the UMLS Semantic Network.

2. **Bilingual term searching**: Each Greek term should have at least one English equivalent (a translation is needed for every Greek term that has not yet been translated). This way, the search in bilingual texts (i.e. English and Greek) would be feasible.

3. **Ontology-based query expansion**: A query that contains a term of a specific semantic type can be enriched with other terms of the same semantic type or with terms of narrower semantic types.
More future work concerning expansion of term extraction and semantic annotation tools as well as enrichment of the linguistic content of our biomedical resources towards bilinguallization, will offer IATROLEXI’s data and tools the chance for further interconnection to multilingual NLP applications in the biomedical domain.

References


1 options:
2 grammar = "Terms";
3 maxdepth = "16";
4 types:
5 MATTRS is set of external
   "com.neurolingo.NLP3.omnium.IRawMorphoAttributes";
6 features:
7 ONTO is object;
8 LEXY is object;
9 functions:
10 HasMAttrs in module LexiconEntry of file internal is method of LEXY ( MATTRS ) as bool;
11 GNC_Agreement in module Agreement of file internal is predicate of ( number, MATTRS ) as object;
12 rules:
13 /* MT_R1 */
14 [IRULE="TERMS_R1"] =>
15 \
16 [LEXY->HasMAttrs([ADJ]), ONTO?=$x:GNC_Agreement(1,[ADJ])],
17 [LEXY->HasMAttrs([N]), ONTO?=$x:GNC_Agreement(1,[N])]
18 /;

Table1. Sample grammar
Figure 1: The annotation process: from raw text input to annotated text output

Figure 2: The ontology construction process