UGTag: morphological analyzer and tagger for the Ukrainian language

Natalia Kotsyba
Warsaw University, Poland
Andriy Mykulyak
A. Sołtan Institute for Nuclear Studies, Warsaw, Poland
Ihor V. Shevchenko
ULIF NASU, Kyiv, Ukraine

Abstract
The UGTag, a program for morphological analysis and tagging of Ukrainian texts is developed within the Polish-Ukrainian Parallel Corpus (PolUKR)[1] project to support morphosyntactic annotation for the Ukrainian part of the corpus. The tagger accepts plain, HTML or XML texts and produces XML files structured according to the XCES standard and suitable for search with such programs as Poliqarp.
The process of the analysis consists of three stages: tokenization, tagging and chunking. At the tokenization stage the text is split into tokens (words, numbers, etc). During the tagging all possible morphological and lemma interpretations are first assigned to each token (morphological analysis), then the correct interpretation is selected (disambiguation). During the chunking stage tokens are grouped into sentences. The Ukrainian Grammatical Dictionary is used as a source of morphological information for the UGTag. It is not restricted to it, however: modular design allows plugging-in additional dictionaries as well as modification of the existing one. Users can interact with the UGTag in three ways: console-based, GUI and Web-based client.

Key words: UGTag, UGD, morphological analyzer, tagger, grammatical dictionary, Ukrainian, Slavic, PolUKR, XCES, corpus.

1. Introduction

UGtag is a set of NLP tools for Ukrainian language. Its development was inspired by a functionally similar TaKIPI[2] toolset for Polish. There are two reasons for this. Firstly, TaKIPI is a very convenient software package with well-thought design that includes all major NLP tasks to prepare an annotated monolingual corpus. Secondly, the UGTag is developed within the Polish-Ukrainian Parallel Corpus project to provide the grammatical annotation for the Ukrainian part of the corpus. Therefore, it is natural to use a unified output format for both language parts of the corpus suitable for search with such programs as Poliqarp[3].
However, some tasks were implemented in a different way. Namely, the UGTag allows for interactive annotation of texts with manual disambiguation and sentencing. Its modular design allows plugging-in additional grammatical dictionaries as well as modification of the existing ones.

2. The program architecture

The UGTag is written in Java, which means that it is platform independent, currently tested under Windows and Linux. It comes in a ready-to-use binary form with no need of compilation of source files.
The scheme below represents the main elements of the program.

[1] This project has received financial support from the Ministry of Science and Higher Education of Poland in 2007-2009. URL: http://corpus.domeczek.pl
[2] All the code for UGTag was written from scratch.
[3] http://korpus.pl/index.php?page=poliqarp.

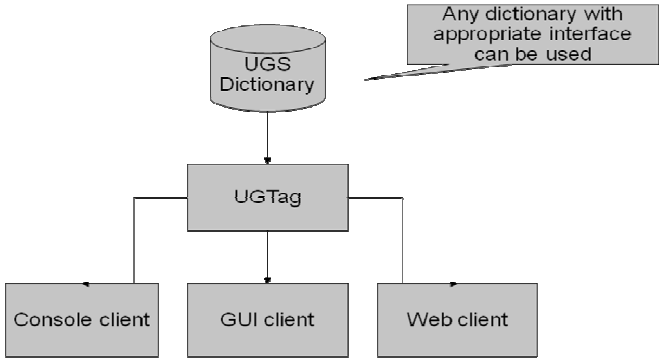

Figure 1. UGTag architecture

Users can work with the program using the console, GUI or a Web client, each of which gives different possibilities. The command-line client provides the fastest access to the basic functionality and is mainly intended for processing large amounts of data.
The GUI client provides some advanced features such as possibility of manual disambiguation, sentence splitting and editing grammatical dictionaries. The Web interface is used for public access to the functionality of the program. It is planned to provide a possibility for users to tag their own texts through the Internet and suggest new words that were not recognized by the program to the general dictionary.
The logic of the program is concentrated in the UGTag package, which is described in more detail below.

3. Dictionaries

The UGTag provides morphosyntactic information to each token in a raw text supplied by the user basing on dictionaries that include a list of word forms and their grammatical interpretations. The core grammatical database is an extended version of the UGD (Ukrainian Grammatical Dictionary). Any compatible dictionary can be used instead, partly because the UGD data are subject to the copyright law.
The Ukrainian Grammatical Dictionary (UGD) was developed in the 1996-2002 at the Ukrainian Linguistic Informational Institute of NASU by Ihor Shevchenko. The UGD contains detailed information about word declination for Ukrainian allowing both morphological analysis and synthesis. The data were initially stored in a relational database and amount to 180 thousand lemmas and 56 thousand endings that can be combined into over 2000 paradigmatic classes.

4. The process of analysis

The UGTag is foreseen to be used for corpora development and includes a number of tasks that can be grouped as follows:

1) pre-processing stage: tokenization and chunking;
2) morphological analysis;
3) disambiguation (leading to proper tagging);
4) sentence grouping.

The following scheme shows in more detail the stages of text processing.

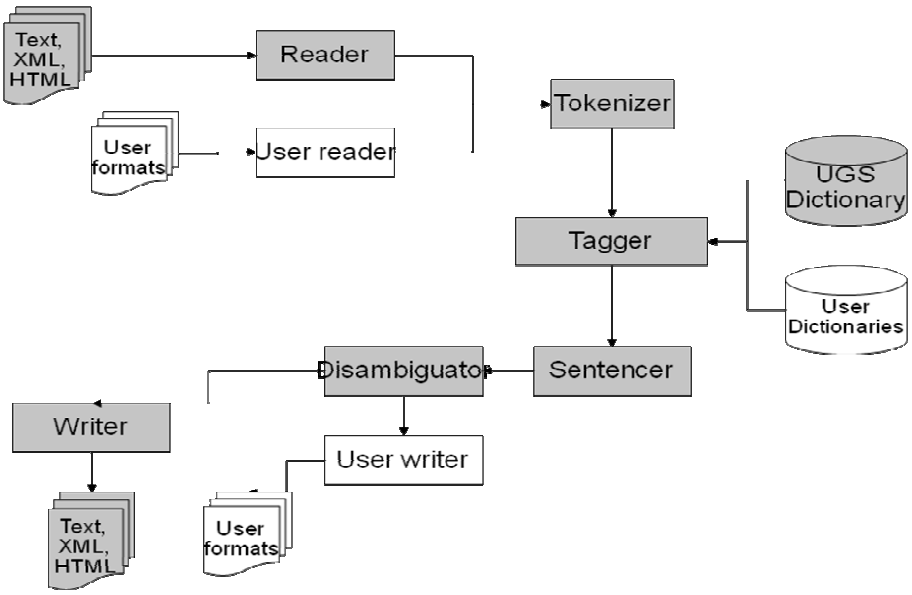

Figure 2. Sequence of logical analysis of corpus texts.

Stage	Role	Input	Output
Reader	Separates (different) external representations of the text from its internal representation (one or more character sequences). In other words, it converts a text to a standard file format.	Texts in different file formats	One or more sequences of characters (usually one sequence per line of the input file)
Tokenizer	Splits sequences into tokens (smallest meaningful pieces of text)	Character sequence	List of tokens
Tagger	Adds morphological information with lemmas to tokens (additionally it can split or group some tokens basing on their meaning – e.g. abbreviations, complex words like „zeleno-červony")	List of tokens	List of tokens with morphological information and lemmas attached to each token
Sentencer	Groups tokens into sentences	List of annotated tokens	List of sentences
Disambiguator	Chooses appropriate grammatical interpretation of the token	List of annotated tokens (optionally augmented with a list of sentences)	List of tokens with a most probable annotation
Writer	Converts list of tokens to the format most appropriate for the user	List of tokens, list of sentences	File with annotations in a specified format

5. Pre-morphological analysis

5.1 Reading phase and input formats

The tagger accepts plain, HTML or XML texts and produces XML files structured according to the XCES standard and suitable for search with such programs as Poliqarp.

By default the program strips all tags from input HTML or XML files and turns them into raw texts. However, users can create and add on the fly to the program their own file readers that take into account the logical mark-up of input XML files and incorporate it into the output XML format. A file reader separates the external representation of texts from their unified internal representation fed to the tokenizer. In other words, it extracts the text itself, possibly portioning it into chunks for further processing.

5.2Tokenization

The pre-morphological analysis presupposes procedures that do not involve the use of the grammatical dictionary.
The tokenizer divides chunks into blocks delimited by whitespace characters. A block can consist of one or more tokens, e.g. a quote and a word with no white space in between (”token). Then it divides blocks into tokens that are minimal structural units. There are five categories of tokens at the moment, more or less corresponding to those in TaKIPI program, namely: words, numbers, punctuation marks, whitespace characters and unrecognized tokens. The word category is defined as a sequence of alphabetical characters with an optional hyphen. If they contain a hyphen, they are classified as technical complex words that are divided into either proper complex words or word collocations during the morphological analysis depending on the existing corresponding records in the grammatical database.

5.3 Morphological analysis and its conceptual basis in the grammatical database

A grammatical dictionary is the core source for the morphological analysis. There are different ways of arranging and presenting grammatical information and very often some additional work has to be carried out to fit a standard or expected representation of grammatical mark-up.
As it was mentioned before, the structure of grammatical information in the UGD was considerably rearranged and further division into finer categories was carried out and implemented to meet the requirements of the intended tagsets. Presently the UGTag supports two tagsets.

5.3.1 The IPIC style tagset

One of the tagsets is basically an extended version of the IPIC[4] tagset that also considers Ukrainian language specific features, see [Kotsyba, Turska, Shypnivska 2008] and [Kotsyba 2009] for details. Some of the changes that were introduced in the Ukrainian grammatical information arrangement to bring the original ULIF tagset (roughly supported by the UGD) to conformity with the IPIC one were the following: the category of degree of comparison for adjectives and adverbs was reintroduced; the category of predicatives was regrouped based on the conclusions in [Derzhanski, Kotsyba 2008].
The notation used in this tagset is also based on the IPIC one but includes a few Ukrainian language specific tags like: adjv (adjectival, a common notion for “syntactic” adjectives like proper adjectives, ordinal numerals and adjectival pronouns), nadj (invariable adjectives), kadj (short form of adjectives), fut (future tense), pres (present tense), numcol (collective numerals)[5].
The POS categorization in the database brings together both the original practical solutions of the UGD and IPIC developers and the traditional intuitive division into parts of speech. For example, the UGD treats nouns of different genders and nouns denoting family names as different parts of speech. Similarly, the IPIC tagset singles out derogative nouns and gerunds. It also groups forms like infinitives, participles, -no/-to forms, etc., as separate word classes[6]. All these were brought together by the cost of partial information redundancy. Basic categories and the part of tag which corresponds to it can be seen in the figure below. They are followed by possible attributes that are ticked off if the category possesses this attribute.

---

[4] IPIC is a shortcut for the IPI PAS corpus, presently the largest publicly available and well-documented corpus for Polish: http://korpus.pl

[5] The Polish grammatical information is slightly altered as well. First of all it concerns the treatment of the so-called predicatives. Secondly, other than personal pronouns were reintroduced as separate categories (in the IPIC they are grouped according to their syntactic features with nouns or adjectives correspondingly without further differentiation). Ordinal numerals are also referred to as a numerals subclass of adjectivals. Untypical adjectives *winien* and *powinien* categorized under “winien” class can be found both by “winien” and “kadj” tags.

[6] These classes are called flexemes which are more tight than the traditional parts of speech and are singled out on the basis of common morphological and syntactic behaviour. The term was introduced by Janusz Bień in the 1960’s.

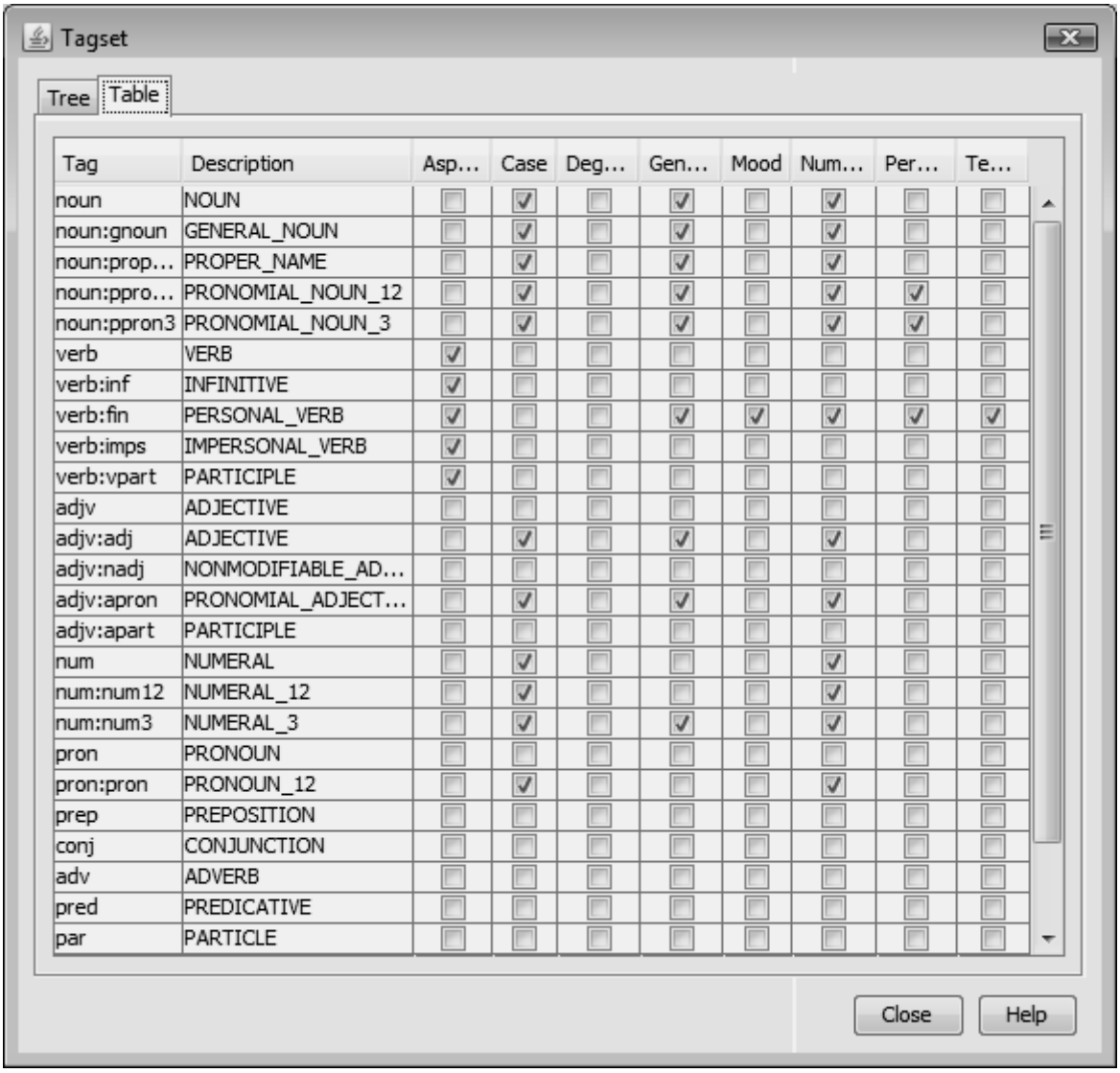

Figure 3. Fragment of the list of categories of the IPIC style tagset.

5.3.1 The MULTEXT-East compatible tagset

The other tagset supported by the UGTag is MULTEXT-East that recently is becoming an international standard. It is foreseen that starting from the version 4, the Ukrainian language will be included into the MULTEXT-East standard[7]. Some theoretical assumptions and decisions that appear in the MTE-4 morphosyntactic specifications for Ukrainian[8] which predefine the corresponding tagset are explained in [Derzhanski, Kotsyba 2009]. One of the major conceptual extensions in the UGTag for the needs of the MTE format is the type of pronouns: they are divided into both syntactic (nominal, adjectival, adverbial) and semantic (demonstrative, indefinite, general, interrogative, reflexive, etc.) types. Conjunctions have acquired the additional attribute of Type: coordinating and subordinating. Prepositions also can be of two types: simple and compound, and are provided with the information about the demanded by them case of the nominal phrase.

It must be noted that all subdivisions that were introduced to meet the requirements of the above described IPIC-like tagset are also used in the MTE one. Thus, the latter is in fact

[7] For the purposes of the PolUKR, a similar package was developed also for the Polish language. A converter from the IPIC-style tagged XML corpus files into MTE-style ones was written to provide the Polish part of the parallel corpus in the MTE tagging format. See [Kotsyba, Radziszewski 2009, to appear] for details.

[8] The draft of the specifications, as well as some of the mentioned papers, are also available at http://domeczek.pl/~natko.

more informative. Nevertheless, the obvious advantage of the IPIC tagset is its intuitive notation which makes searching through the corpus much easier and does not demand either memorizing the tags or creating a sophisticated user interface with hints for search.

5.3.3 Changes in the characteristics of the language material

In order to satisfy the described tagsets, relevant information had to be added to the dictionary database, making it even a richer resource than the original UGD. At the same time, the changes that were made influenced the number of lexemes in the main dictionary. For example, adjectives and adverbs of comparative and superlative degrees that were presented as separate lemmas in the UGD were enriched with the degree information and relemmatized accordingly. This led to the decrease of number of adjectives in the dictionary by ca. 2200 words. This procedure was partly automated basing on typical suffixes and prefixes for the comparative and superlative degrees and a list of exceptions. The correct lemmas were deduced basing on the similarity of stems and morphological rules of degree transformation. The results were checked manually.

158 prepositions were supplied with the information about the case they govern. Those that govern several cases are treated as homonyms, which increased their general number by ca. 30.

Adjectival participles, treated as verb forms in the UGD, were extracted and modified to present separate lemmas. Ca. 20 paradigmatic classes were defined for them. The number of newly acquired word forms was quite significant due to the introduced feminine, neutral and plural forms, as well as five new cases for each gender and number, that were absent in the UGD.

Pronouns were divided into semantic types (the Type attribute in the MTE tagset). In cases when more than one semantic feature was presented in one pronoun, the second one was recorded as a Referent_Type. The word *чийсь* („somebody's") is an example of combination of an indefinite and a possessive type[9].

Some other changes concern word splitting. Originally the UGD contains collocations with white space characters or hyphens treated as individual units. During the tokenization phase, the former ?ones are split into separate words and they are not recognized by the analyzer as a whole[10]. A highly productive group with a hyphen combining two or more adjectives that is not systematically represented in the UGD[11], is treated in the UGTag space as combinations of separate words. Most of the rest hyphen-containing words were left intact.

5.3.4 Working with the dictionary filter

Information used for further grammatical mark-up can be sorted for various kinds of queries and correctness control with the help of the dictionary filter. It allows setting restrictions on the category and searching for lemmas with the help of regular expressions, cf. the figure below.

[9] This way of representation in the MTE-4 was selected for more conformity with the existing MTE notation for other languages, see [Derzhanski, Kotsyba 2009] for details.

[10] However, information about those combinations is preserved and can be used for syntactic analysis in the future.

[11] It is practically impossible to cover the whole class as it is highly productive.

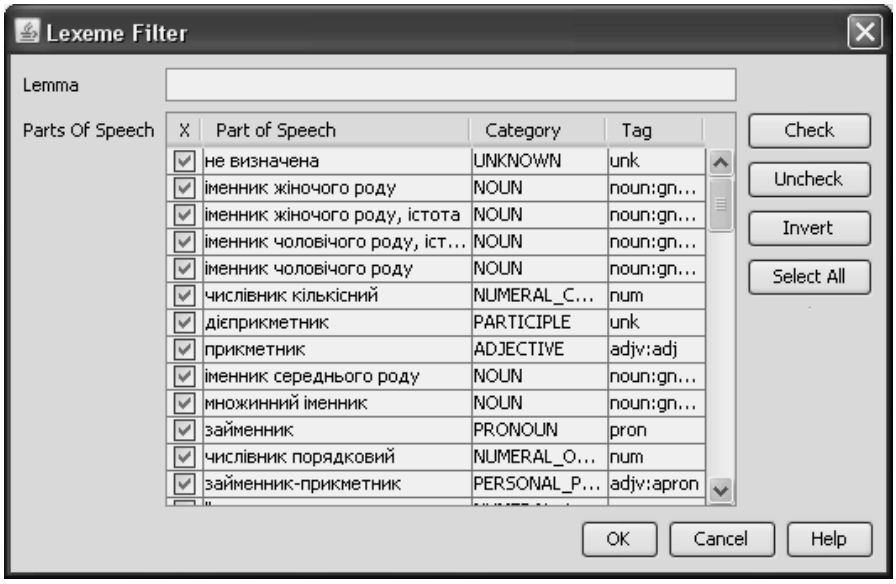

Figure 4. Lexeme and category filter

The result of search applying the dictionary filter for the class of feminine nouns:

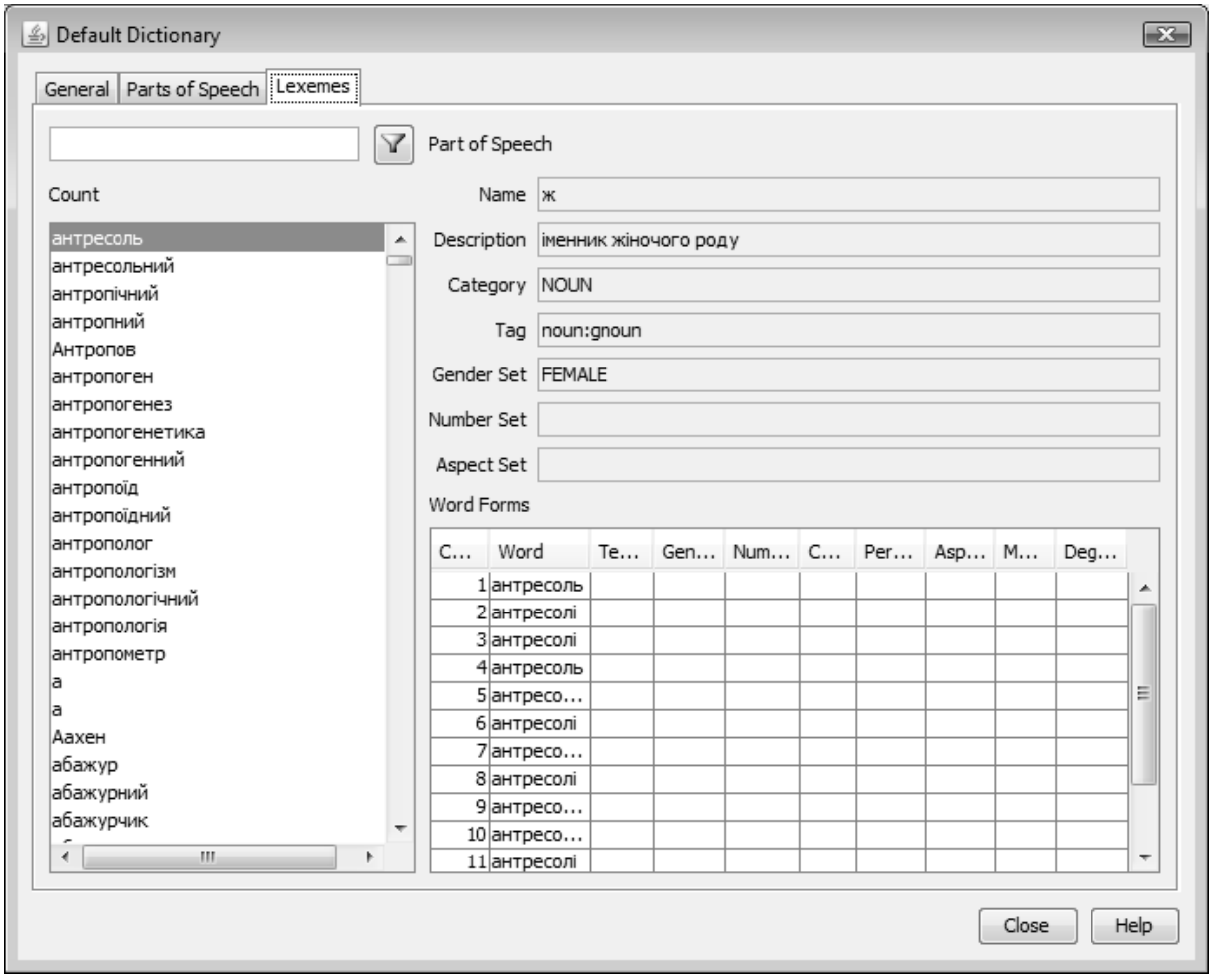

Figure 5. The content of the default grammatical dictionary

The window includes a search line for typing in a lemma or its part (regexp). The “count” parameter shows the quantity of results that meet the requirements of search and below is the list of those lemmas in the alphabetical order.
On the right side, the morphological features of the selected class are shown in corresponding fields. A selected from list lemma appears with its full paradigm in the right bottom corner. All available grammatical values for each form are shown as well.
These features of the UGTag are mainly useful for grammatical reference and the dictionary content control.

5.4 Sentencing

Sentence splitting as part of structural text mark-up is done after grammatical annotation because it is rule-based and some of those rules require grammatical information. The implemented so far rules are partially based on Rudolf’s work for Polish [Rudolf 2004].[12]
The rules are an interplay of heuristics that use popular abbreviations and words starting with the capital letter, whose meaning is also taken into the account. More work on enhancing automated structural mark-up is planned in the nearest future.

5.5 Writing phase and writing format

At the moment we foresee two output tag formats for resulting XML files. The default format is based on the TaKIPI one (version 1.8) for Polish but extended for Ukrainian specific features, see [Kotsyba, Turska, Shypnivska 2008], slightly modified. It retains maximum grammatical information that can be provided by the Polish and Ukrainian grammatical dictionaries. The second available format is MULTEXT-East compatible. As well as in the case of file readers, users can define their own writers that produce output files with customized mark-up.

6. Working with texts

The choice of the intended tagset is suggested to the user immediately after loading the text file for analysis and can be changed during the process of work.
The UGTag accepts raw texts or structured HTML or XML files. The structural information (division into paragraphs and sentences) can be retained or replaced by the UGTag.
The user can watch the progress of tagging as it goes. Tagged tokens of different categories are displayed in the screen colour coded[13]. For example, in the window below the red colour[14] marks unrecognized tokens, e.g. “№14”, the green one marks words with only one available grammatical interpretation (this is important because these do not need disambiguation afterwards), the blue one shows words with multiple grammatical interpretations. They are additionally marked by the italic in this case.

[12] The implementation of the sentence-splitting algorithm was carried out by Oresta Tymchyshyn.
[13] Colour coding can be changed by the user through the configuration menu.
[14] The original figure is coloured.

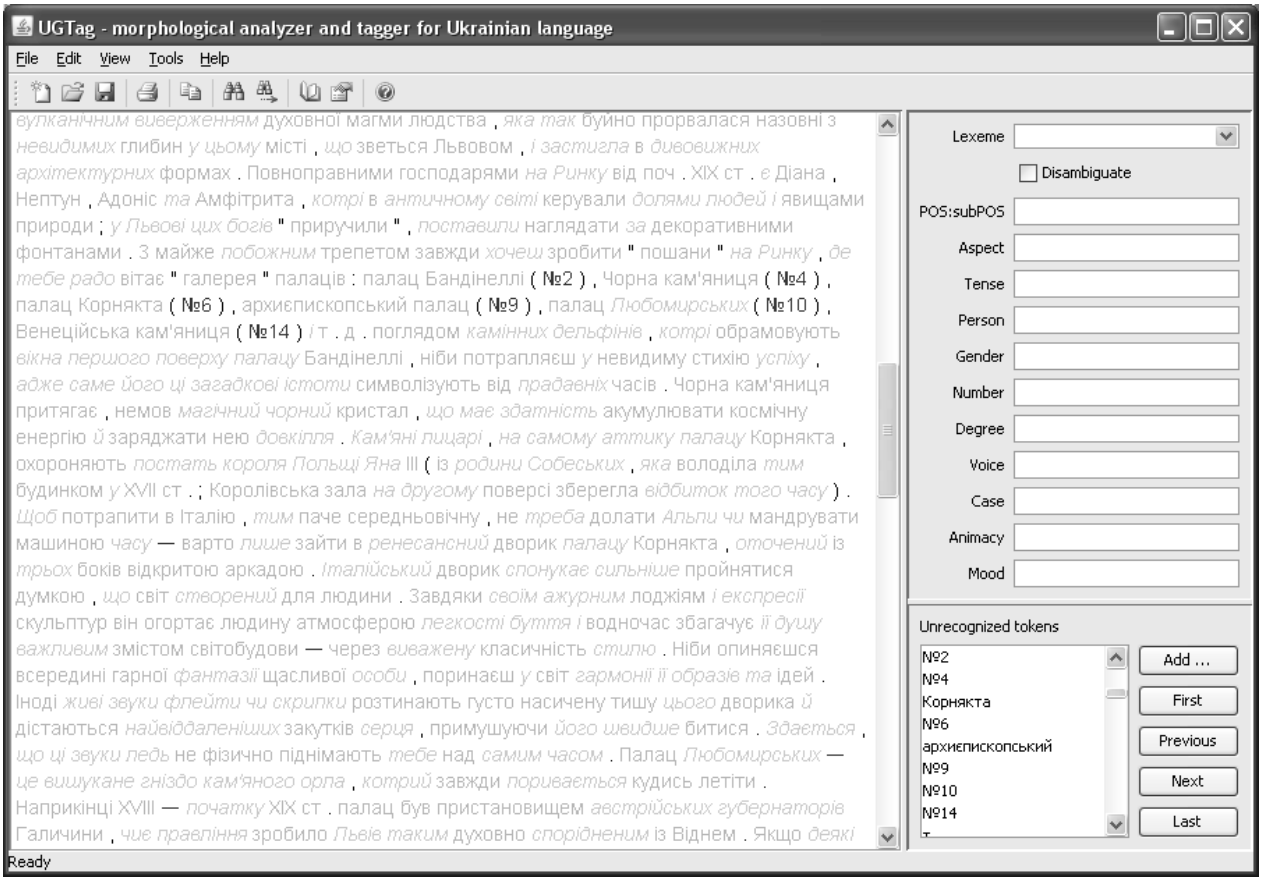

Figure 5. Text processing.

The panel in the top right corner displays grammatical characteristics of the selected item and the possibility of manual disambiguation is given for words with multiple available interpretations.
The bottom right corner displays a list of words that were not unrecognized by the active built-in dictionary. The user can select a word from this list and add it to the common bulk of words by clicking the “add” button which instantiates a further dialog.

6.1 Automatic disambiguation

Preliminary rules for automatic disambiguation based on statistical analysis were devised for a small but frequently used word class of prepositions. For example, the word “do” can have 15 grammatical interpretations, only one of which is preposition “do” and the rest are all possible interpretations of the invariable noun “do” (a musical note) The tagger chooses the prepositional interpretation at the moment as the most frequent one. Also, the lexeme “na” is most frequently used in the prepositional function, although the colloquial use of it as an interjection is possible as well. Further disambiguation policy foresees combination of rules and statistical analysis of manually disambiguated data as training samples for machine learning.

6.2 Ways of enriching the dictionary database

During annotation the UGTag automatically creates a list of words that were not found in the dictionary and displays it to the user and allows adding them to one of custom dictionaries. Custom dictionaries can be used along with the default one or instead of it, enhancing the quality of annotation.
The lemma for a new word should be introduced manually (the selected input word form is prompted by default) and the part of speech should be selected from the drop-down list.

The program gives hints as to the paradigm of the word. The grammatical paradigm class identifier is assigned to the new word based on the answers provided by the user. The most probable grammatical paradigm for the selected part of speech is generated in a table below and the user is prompted to confirm it in case it is correct for the given word. Other declension paradigms can be accessed by navigating if the first one fails, or definition of the word forms can be done manually.
This feature is available through the GUI client and will be also available through Web client.
Users can also create their own dictionaries by using the embedded dictionary editor.
Other ways of extending the lexical and grammatical database for tagging can be developed based on various heuristics, e.g. for derivational patterns, if this will be justified experimentally.

7. Plans for further development

Extensive experimenting with real corpus texts will indicate the directions for further development of the program. One of our priorities is enriching the dictionary database using both manual and automatic ways as well as enhancing the quality of automatic disambiguation. We also plan to concentrate on word grouping for syntactic analysis, including first of all complex words like numerals: “dvadciat’ try” (twenty three) currently recognized as separate words (“dvadciat’” and “try”), complex passive structures, prepositional phrases, etc.

Bibliography

1. Broda B., Piasecki M. and Radziszewski A. (2008). Towards a Set of General Purpose Morphosyntactic Tools for Polish. *Proceedings of Intelligent Information Systems, Zakopane, Poland, 2008*. Institute of Computer Science—PAS.
2. Derzhanski I. and Kotsyba N. (2008). The category of predicatives in the light of the consistent morphosyntactic tagging of Slavic languages. In *Lexicographic Tools and Techniques: Proceedings of the MONDILEX First Open Workshop*, pages 68–79, Moscow: IITP—RAS.
3. Derzhanski I. and Kotsyba N. (2009). *Towards a Consistent Morphological Tagset for Slavic Languages: Extending MULTEXT-East for Polish, Ukrainian and Belarusian.* Metalanguage and Encoding Scheme Design for Digital Lexicography: MONDILEX Third Open Workshop, Bratislava, 15–16 April 2009.
4. Kotsyba N. (2009). *The Current State of Work on the Polish-Ukrainian Parallel Corpus (PolUKR).* Proceedings of the International Workshop within MONDILEX project "Problems of Slavic Lexicography" Kyiv, 2-4 February 2009.
5. Kotsyba N., A. Radziszewski (to appear). Integrating the Polish language into the MTE family, Ljubljana.
6. Kotsyba N., M. Turska, O. Shypnivska (2008). *Linguistic principles of organizing a common morphological tagset for PolUKR (Polish-Ukrainian Parallel Corpus).* In *Proceedings of the international conference* "Intelligent Information Systems, 16-18 June 2008, Zakopane, Poland".
7. Rudolf, Michał (2004). *Metody automatycznej analizy korpusu tekstów polskich*. Warszawa: Uniwersytet Warszawski – Wydział Polonistyki.
8. Shevchenko I. V. (1996). Algorytmična slovozminna klasyfikacija ukrajins’koji leksyky. *Movoznavstvo.* Vol. 4–5, Kyiv, p. 40–44.
9. Shevchenko I. V., Shirokov V.A., Rabulets’ A.G. (2005). Elektronnyj grammatičeskij slovar’ ukrainskogo jazyka. In: *Proceedings of the international conference „Megaling’2005. Prikladnaja lingvistika v poiske novyx putej”, 27.VI–2.VII 2005*. Meganom, Crimea, Ukraine, p. 124–129.

10. Shevchenko Ihor (2008). Parametryzacija jak osnova hramatyčnoji identyfikaciji slovnykovyx odynyc' ukrajins'koji movy. In: *Prykladna lingwistyka ta lingvistyčni texnolohiji. Megaling-2007. Zbirnyk naukovyx prac'*. Kyiv: Dovira, p. 393–402.
11. Shirokov V. A., Rabulets' O. H., Shevchenko I. V. , Kostyshyn O. M., Yakymenko K. M. (2007). *Intehrovana leksykohrafična systema „Slovnyky Ukrajiny"*, version 3.1. Kyiv. CD-edition.