System of Intellectual Ukrainian Language Processing

© Nataliia Tmienova [1] and © Bogdan Sus' [2]

[1]Faculty of Information Technology Taras Shevchenko National University of Kyiv, Ukraine
[2]Institute of High Technologies Taras Shevchenko National University of Kyiv, Ukraine

tmyenovox@gmail.com,bnsuse@gmail.com

Abstract. The problem of high-quality automatic natural language processing is one of the most important problems in computational linguistics. Automatic natural language processing is used in information retrieval, in tasks of text generation and text recognition, in machine translation, in sentiment analysis and so on. All of these areas require specialized linguistic and mathematical models to represent the morphology, syntax, and semantics of text in a form that is convenient for automatic processing.

The article describes a developed system that implements specific linguistic tasks related to the processing of Ukrainian language, that is text preprocessing, morphological and lexical analyzes of text. In order to create such a system, an analysis of the available natural language text-processing tools was carried out and the possibility of using them for text processing of Ukrainian language was examined. Also, the most appropriate text processing tools in Ukrainian language were selected. The basic stages of text preprocessing were considered in detail and algorithms of their program implementation were given. In addition, the results of the developed system were demonstrated.

Keywords: Natural language processing, NLP, Text Mining, Ukrainian language processing, language analysis, text corpus

1 Introduction

Natural language processing (NLP) is a subfield of computer science, information engineering, and artificial intelligence concerned with the interactions between computers and human (natural) languages, in particular how to program computers to process and analyze large amounts of natural language data [1].

NLP is a component of text mining that performs a special kind of linguistic analysis that essentially helps a computer “to read” text**.** NLP uses a variety of methodologies to decipher correctly the ambiguities in human language, including the following: automatic summarization, part-of-speech tagging, disambiguation, entity extraction and relations extraction, as well as natural language understanding and recognition [2].

Today, NLP software is a “shadow” process running in the background of many common applications such as the personal assistant features in smartphones, translation software and in self-service phone banking applications. NLP is used in information retrieval, in tasks of text generation and text recognition, in sentiment analysis and so on. All of these areas require specialized linguistic and mathematical models to represent the morphology, syntax, and semantics of text in a form that is convenient for automatic processing.

For the correct functioning of natural language processing software it is necessary to use a consistent knowledge base such as a detailed thesaurus, a lexicon of words, a data set for linguistic and grammatical rules, an ontology and up-to-date entities, text corpora.

It is now quite difficult to imagine linguistic researches, language learning and the translation process without the use of corpora [3, 4, 5]. Corpora are widely used in lexicographic and grammatical studies, semantics and stylistics. The term "linguistic corpus" means the electronic collection of natural language texts, which is organized and designed in a certain way and is intended for the scientific and practical study of language [6].

Data collected in the corpus can be very different in quality and quantity depending on the project of the investigation [7]. So, the formation of a corpus of any language texts should begin with the clarification of the research and educational tasks that are supposed to be solved on the basis of the content of the created corpus. The purpose of the corpus determines the features of its structure. Corpus can be considered as an important means of verifying a variety of linguistic theories. Therefore, the corpus must be representative and balanced. Accordingly, linguistic tools for corpus work are developed with aim at performing a particular task.

The major problem of natural language processing is the ambiguity, so most natural language processing problems can be considered as finding proper interpretation. According to the structure of natural language, we can distinguish the following basic stages of linguistic analysis, each of which is ambiguous in its own way: previous, morphological, syntactic and semantic. These stages of analysis are performed sequentially and each subsequent stage uses the results of the previous ones. Similarly, mistakes in the previous stages of analysis are affected by the results of the following stages.

2 The purpose of the article

There are many different types of systems that implement the steps of text analysis described above in English and Russian (AOT, Rosette text analytics, Polyglot). At the time, the development of a system for processing Ukrainian language texts is a rather urgent problem, because, unfortunately, there are not enough tools for Ukrainian language processing, namely: libraries for programming languages, marked corpora, dictionaries, thesauruses, etc.

The purpose of the development of a system for intellectual Ukrainian language processing was a creation of tools for the processing of natural language texts in Ukrainian at preliminary, morphological and lexical levels of analysis.

The main objectives of the preliminary analysis (text preprocessing) include the following:

- tokenization is the process of splitting the text into units called tokens (tokens can be words, numbers, punctuation marks, etc.);
- sentence boundary detection.

It is also desirable to solve at this stage the following minor problems:

- removing of non-text elements (tags, meta-information);
- removing formatting (italics, underline, bold);
- selection of e-mail addresses;
- selection of file names;
- assembly of words written with letter-spacing;
- removing of stop words;
- named entity recognition.

The tasks of morphological analysis include the following:

- definition of the grammatical attributes of the word (defining of the parts of the speech (POS) and the grammatical categories that are inherent in the corresponding POS - number, gender, case, etc.);
- stemming is the process of reducing words consisting of several morphemes to their stem, i.e. to the fixed basis;
- lemmatization is the process of reducing words to their vocabulary form.

Among the problems of lexical analysis there are the following:

- defining unique words;
- determining the frequency of words;
- calculating the lexical diversity.

In the process of literature analyzing, such a complex system, which would cover the processing of Ukrainian texts at several levels, was not revealed. Only some separate language processing tools have been found. After testing, it turned out that some of them produce incorrect results. There was also an attempt to adapt the stemming algorithm for Russian language [8] to Ukrainian language, but the results of this algorithm were not acceptable.

The NLTK library tools were selected for the preliminary analysis [9]. On this basis, tools for solving the following problems of preliminary analysis of Ukrainian texts such as tokenization, sentence boundary detection, removing of non-text elements (tags, meta-information), e-mail highlighting, selection of file names, assembly of words written with letter-spacing, removing of stop words, named entity recognition were developed.

The pymorphy2 library was selected as the basis of the morphological analyzer [10]. Pymorphy2 uses a large electronic dictionary of Ukrainian language [11] converted to the OpenCorpora format [12]. On the basis of the tools of this library, tools for the realization of the following tasks of morphological analysis such as morphological analysis of a word, stemming, lemmatization were created.

The simple lexical analysis was developed using Python tools. The matplotlib library was used to display the lexical information on the graph. For lexical analysis of Ukrainian texts, the system contains the following tools: calculating word frequencies and displaying them on a graph; calculating the lexical diversity of the text.

3 Available NLP tools review

Modern tools for text analysis can be divided into two major categories:

- specialized tools are tools for language-specific analysis (morphological analyzers, syntax parsers, etc.);

- integrated packages are software instruments that provide features for analyzing the text at different levels.

Let us describe several platforms and libraries for natural language text processing. The following software products provide tools for analyzing texts in natural language, both at one level and at many levels.

The Rosette text analytics platform develops software [13] to extract information from unstructured text for use by search engines and data analytics applications.

For basic text analysis, the platform provides specific language tools for tokenization, selection POS, lemmatization, classification, named entity recognition, and relations between named entities. For text preprocessing and morphological analysis, the platform provides the following features:

1) sentence boundary detection:

- figures on ambiguous punctuation marks for abbreviations, file names, email addresses, etc;
- uses machine learning and statistical analysis;
- supports Ukrainian;

2) tokenization:

- uses statistical modeling;

3) morphological analysis:

- determines POS for ambiguous words by the means of statistical modeling for lemmatization;
- decomposes difficult words;
- uses lemmatization for stemming;
- does not support Ukrainian.

Polyglot [14] is a library on Python for natural language processing. For text preprocessing and morphological analysis it contains the following modules:

1) tokenization module:

- in addition to the tokenization itself, it has tools for splitting text into sentences;
- supports Ukrainian;
- dividing into sentences does not take into account Ukrainian cuts;

2) POS definition:

- does not support Ukrainian language;

3) dividing words into morphemes:

- incorrectly divides Ukrainian words.

AOT system (Automatic text processing) [15] is a set of packages designed for text processing in Russian. It contains separate components named processors for text preprocessing and morphological analysis.

The components that form up the language model are the linguistic processors that process the input text by the conveyor method. The input of one processor is the output of another. The following components are distinguished:

- text preprocessing;
- morphological analysis;
- parsing;
- semantic analysis.

The text preprocessing processor contains an algorithm for splitting into tokens and sentences.

The morphological processor uses a special Russian morphological dictionary based on the A.A. Zaliznyak grammar dictionary. It includes 161000 lemmas.

In case of lemmatization, a lot of morphological interpretations of the following form are given for each input word:

1) lemma;
2) POS;
3) grammatical categories of the word.

Natural Language Toolkit Library or NLTK [9] is a software package for symbolic and statistical processing of natural language in Python. It contains graphical representations and examples of data. It is accompanied by extensive documentation, including a book explaining the basic concepts behind the natural language processing tasks that can be accomplished with this package [16].

NLTK is an optimal software platform for prototyping and development of linguistic research systems.

NLTK supports classification, tokenization, POS defining, syntactic analysis.

Key features:

- lexical analysis: a tokenizer of words and texts;
- n-grams and word combinations;
- POS defining;
- lemmatization;
- stemming;
- named entity recognition.

NLTK is a powerful library for English. The morphological and syntactic analyzers do not support Ukrainian, but the pymorphy2 module is available for morphological analysis of the Russian and Ukrainian languages.

Pymorphy2 Morphological Analyzer [10] is an open-source Python programming library for morphological analysis of Russian and Ukrainian words.

Main functionality:

- provides information on basic grammatical categories;
- declines words for the case;
- puts the word in its original form.

In pymorphy2, the MorphAnalyzer class is used for morphological analysis of words.

4 The system structure, methods used to the development of the system, description of implementation

The structure of the system can be represented as the following flowchart (Fig.1).

Regular expressions are used to solve a large number of subtasks. A regular expression is a special text string that describes or matches multiple lines according to a set of special syntax rules (that is, a search pattern). They are used in many text editors and auxiliary tools to find and modify text based on specified templates.

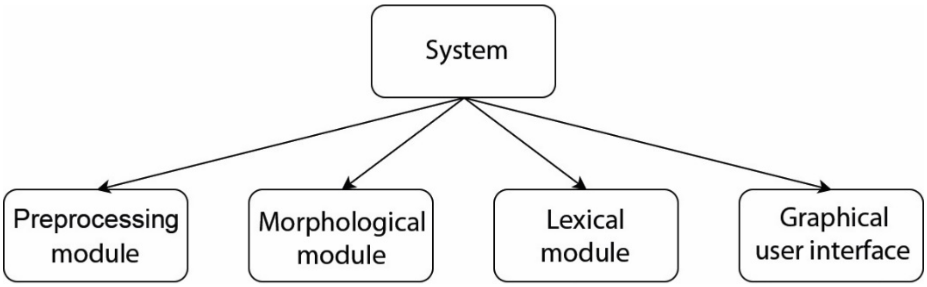


Fig. 1. Structure of system.

The NLTK library was selected for the tokenization and solving of sentence boundary problems. Because it is necessary to consider abbreviations to determine the boundaries of words and sentences, a tokenizer of the NLTK library that works with regular expression patterns was used and a list of abbreviations was made using DSTU 3582-97 [17]. Regular expression was used for tokenization:

```
[^,.;:!?\s]+(?:\.[^ЙЦУКЕНГШЩЗХЇҐФІВАПРОЛДЖЄЯЧСМИТЬБЮAZ\s])*[^,;.:!
?\s]*|[,;.:!]
```

The boundaries of sentences are determined by regular expression

```
.*?(?:\S{2,}|\s)[.!?](?![йцукенгшщзххїґфівапролджєячсмитьба-z])
```

Regular expressions were used to highlight non-text items, e-mail addresses, and filenames.

The removing of stop words was performed using a stop word list.

A recursive algorithm that uses a dictionary of words was developed to minimize the quantity of words written with letter-spacing.

The pymorphy2 bibliography was used to obtain the morphological information about the word. Based on information about the lemma and word form, stemming is performed.

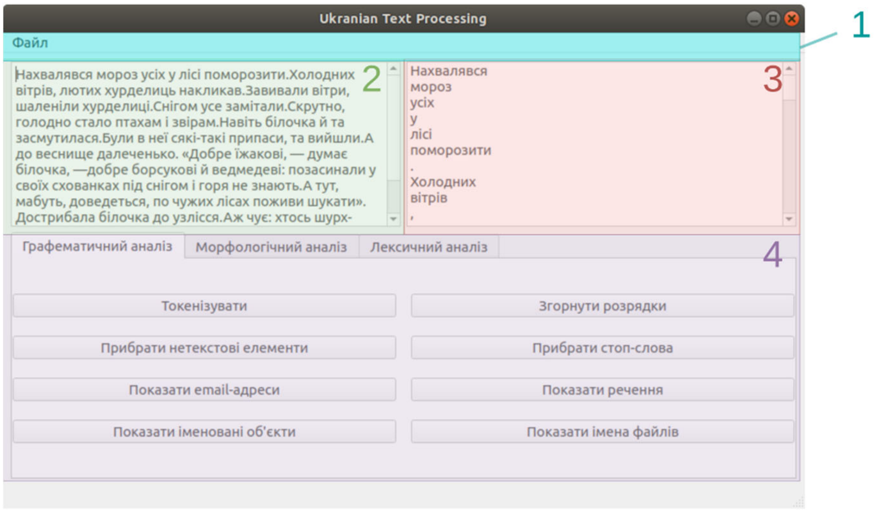
The search for named entities in the text is based on a list of named words based on the NER annotation of Ukrainian corpus [18].

The simple lexical analysis was written on Python. The matplotlib library was used to display the lexical information on the graph.

Python was chosen to implement the software system. The program's GUI is written using the PyQt GUI. A matplotlib library that is a Python library for building 2D graphs that creates shapes in a variety of print formats and interactive environments across platforms was used to build the graphs.

5 Demonstration of results

The GUI of the system consists of four areas, which are shown in Fig. 2.

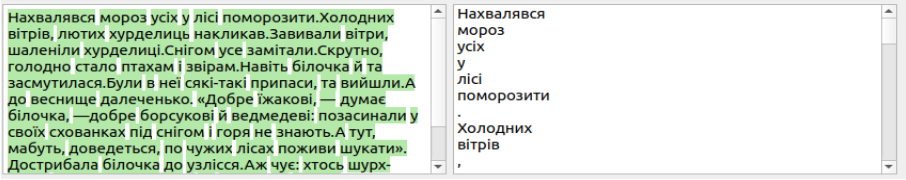


```
Нахвалявся мороз усіх у лісі поморозити.Холодних
вітрів, лютих хурделиць накликав.Завивали вітри,
шаленіли хурделиці.Снігом усе замітали.Скрутно,
голодно стало птахам і звірам.Навіть білочка й та
засмутилася.Були в неї сякі-такі припаси, та вийшли.А
до веснище далеченько. «Добре їжакові, — думає
білочка, —добре борсукові й ведмедеві: позасинали у
своїх схованках під снігом і горя не знають.А тут,
мабуть, доведеться, по чужих лісах поживи шукати».
Дострибала білочка до узлісся.Аж чує: хтось шурх-
```

```
Нахвалявся
мороз
усіх
у
лісі
поморозити
.
Холодних
вітрів
,
```

Fig. 2. 1 - menu area, 2 - input area, 3 - token list area, 4 - language processing feature set area

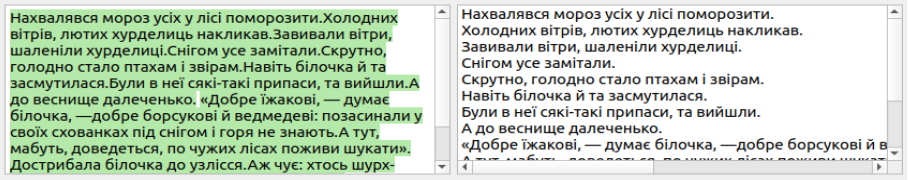
The results of the tokenization and sentence splitting of the input text are shown in Fig. 3 and Fig. 4 accordingly.



```
Нахвалявся мороз усіх у лісі поморозити.Холодних
вітрів, лютих хурделиць накликав.Завивали вітри,
шаленіли хурделиці.Снігом усе замітали.Скрутно,
голодно стало птахам і звірам.Навіть білочка й та
засмутилася.Були в неї сякі-такі припаси, та вийшли.А
до веснище далеченько. «Добре їжакові, — думає
білочка, —добре борсукові й ведмедеві: позасинали у
своїх схованках під снігом і горя не знають.А тут,
мабуть, доведеться, по чужих лісах поживи шукати».
Дострибала білочка до узлісся.Аж чує: хтось шурх-
```

```
Нахвалявся
мороз
усіх
у
лісі
поморозити
.
Холодних
вітрів
,
```

Fig. 3. The result of tokenization

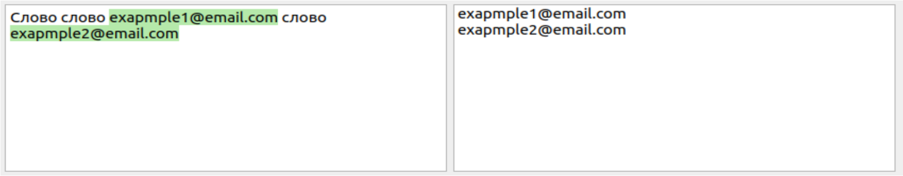


```
Нахвалявся мороз усіх у лісі поморозити.Холодних
вітрів, лютих хурделиць накликав.Завивали вітри,
шаленіли хурделиці.Снігом усе замітали.Скрутно,
голодно стало птахам і звірам.Навіть білочка й та
засмутилася.Були в неї сякі-такі припаси, та вийшли.А
до веснище далеченько. «Добре їжакові, — думає
білочка, —добре борсукові й ведмедеві: позасинали у
своїх схованках під снігом і горя не знають.А тут,
мабуть, доведеться, по чужих лісах поживи шукати».
Дострибала білочка до узлісся.Аж чує: хтось шурх-
```

```
Нахвалявся мороз усіх у лісі поморозити.
Холодних вітрів, лютих хурделиць накликав.
Завивали вітри, шаленіли хурделиці.
Снігом усе замітали.
Скрутно, голодно стало птахам і звірам.
Навіть білочка й та засмутилася.
Були в неї сякі-такі припаси, та вийшли.
А до веснище далеченько.
«Добре їжакові, — думає білочка, —добре борсукові й в
```

Fig. 4. The result of the sentence splitting

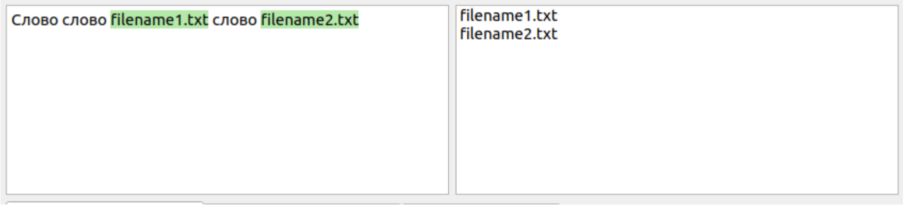
The functions and results of text preprocessing are shown in Fig. 5-9.



Слово слово exapmple1@email.com слово exapmple2@email.com

exapmple1@email.com
exapmple2@email.com

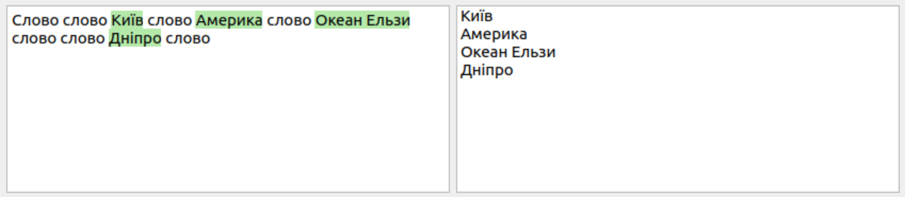
Fig. 5. Selection of e-mails



Слово слово filename1.txt слово filename2.txt

filename1.txt
filename2.txt

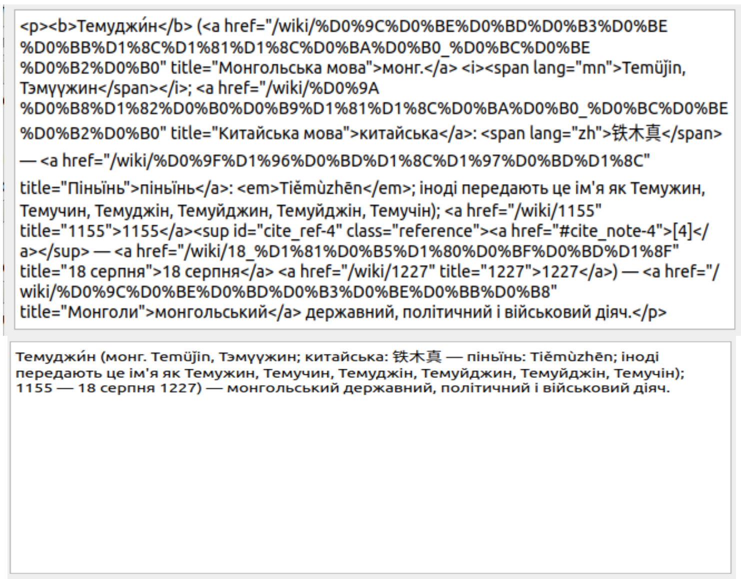
Fig. 6. Selection of file names



Слово слово Київ слово Америка слово Океан Ельзи слово слово Дніпро слово

Київ
Америка
Океан Ельзи
Дніпро

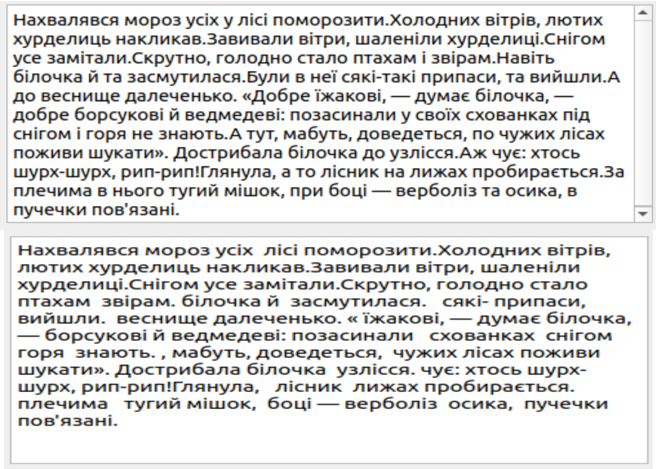
Fig. 7. Selection of named entity



```
<p><b>Темуджи́н</b> (<a href="/wiki/%D0%9C%D0%BE%D0%BD%D0%B3%D0%BE
%D0%BB%D1%8C%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE
%D0%B2%D0%B0" title="Монгольська мова">монг.</a> <i><span lang="mn">Temüjin,
Тэмүүжин</span></i>; <a href="/wiki/%D0%9A
%D0%B8%D1%82%D0%B0%D0%B9%D1%81%D1%8C%D0%BA%D0%B0_%D0%BC%D0%BE
%D0%B2%D0%B0" title="Китайська мова">китайська</a>: <span lang="zh">铁木真</span>
— <a href="/wiki/%D0%9F%D1%96%D0%BD%D1%8C%D1%97%D0%BD%D1%8C"
title="Піньїнь">піньїнь</a>: <em>Tiěmùzhēn</em>; іноді передають це ім'я як Темужин,
Темучин, Темуджін, Темуйджин, Темуйджін, Темучін); <a href="/wiki/1155"
title="1155">1155</a><sup id="cite_ref-4" class="reference"><a href="#cite_note-4">[4]</
a></sup> — <a href="/wiki/18_%D1%81%D0%B5%D1%80%D0%BF%D0%BD%D1%8F"
title="18 серпня">18 серпня</a> <a href="/wiki/1227" title="1227">1227</a>) — <a href="/
wiki/%D0%9C%D0%BE%D0%BD%D0%B3%D0%BE%D0%BB%D0%B8"
title="Монголи">монгольський</a> державний, політичний і військовий діяч.</p>
```

Темуджи́н (монг. Temüjin, Тэмүүжин; китайська: 铁木真 — піньїнь: Tiěmùzhēn; іноді передають це ім'я як Темужин, Темучин, Темуджін, Темуйджин, Темуйджін, Темучін); 1155 — 18 серпня 1227) — монгольський державний, політичний і військовий діяч.

Fig. 8. Selection of non-text elements



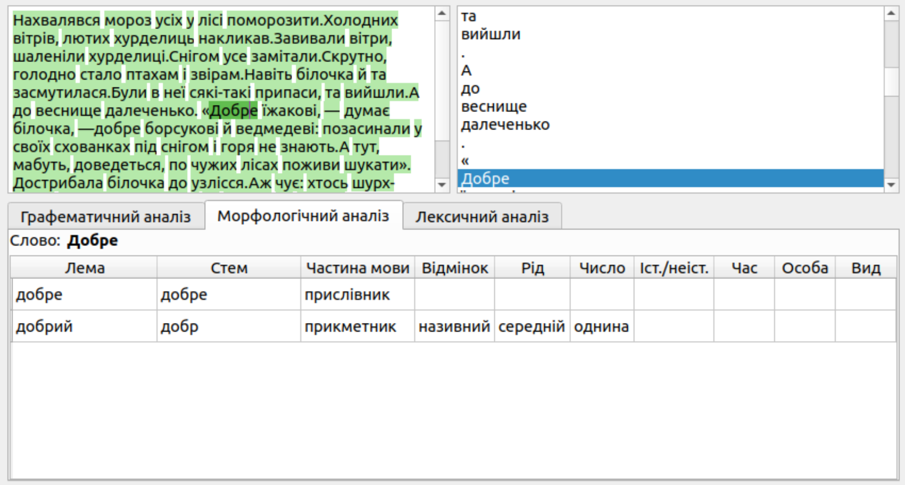
Нахвалявся мороз усіх у лісі поморозити.Холодних вітрів, лютих хурделиць накликав.Завивали вітри, шаленіли хурделиці.Снігом усе замітали.Скрутно, голодно стало птахам і звірам.Навіть білочка й та засмутилася.Були в неї сякі-такі припаси, та вийшли.А до веснище далеченько. «Добре їжакові, — думає білочка, — добре борсукові й ведмедеві: позасинали у своїх схованках під снігом і горя не знають.А тут, мабуть, доведеться, по чужих лісах поживи шукати». Дострибала білочка до узлісся.Аж чує: хтось шурх-шурх, рип-рип!Глянула, а то лісник на лижах пробирається.За плечима в нього тугий мішок, при боці — верболіз та осика, в пучечки пов'язані.

Нахвалявся мороз усіх лісі поморозити.Холодних вітрів, лютих хурделиць накликав.Завивали вітри, шаленіли хурделиці.Снігом усе замітали.Скрутно, голодно стало птахам звірам. білочка й засмутилася.   сякі- припаси, вийшли. веснище далеченько. « їжакові, — думає білочка, — борсукові й ведмедеві: позасинали   схованках снігом горя знають. , мабуть, доведеться, чужих лісах поживи шукати». Дострибала білочка узлісся. чує: хтось шурх-шурх, рип-рип!Глянула,   лісник лижах пробирається. плечима   тугий мішок, боці — верболіз осика, пучечки пов'язані.

Fig. 9. Removing of stop-words.

For morphological analysis, tokenization must be carried out in advance. To extract morphological information for a word from a text, it is necessary to click on the desired word in the text or the list on the right.

The morphological analysis of the selected word from the text is shown in Fig. 10.



Нахвалявся мороз усіх у лісі поморозити.Холодних вітрів, лютих хурделиць накликав.Завивали вітри, шаленіли хурделиці.Снігом усе замітали.Скрутно, голодно стало птахам і звірам.Навіть білочка й та засмутилася.Були в неї сякі-такі припаси, та вийшли.А до веснище далеченько. «Добре їжакові, — думає білочка, —добре борсукові й ведмедеві: позасинали у своїх схованках під снігом і горя не знають.А тут, мабуть, доведеться, по чужих лісах поживи шукати». Дострибала білочка до узлісся.Аж чує: хтось шурх-

та
вийшли
.
А
до
веснище
далеченько
.
«
Добре

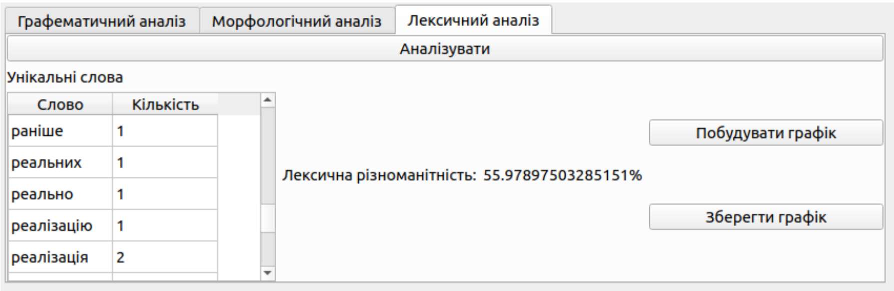
Слово: **Добре**

Лема	Стем	Частина мови	Відмінок	Рід	Число	Іст./неіст.	Час	Особа	Вид
добре	добре	прислівник							
добрий	добр	прикметник	називний	середній	однина				

Fig. 10. The window tab of morphological analysis

Morphological analysis is shown in a table where rows are possible parsing words and columns are grammatical categories. The lexical analysis tab and the lexical characteristics of a news article are presented in Fig. 11.

A graphical representation of the lexical diversity of the text is shown in Fig. 12. To save the received graphs, it is necessary to click on the button “Save graph”.



Унікальні слова

Слово	Кількість
раніше	1
реальних	1
реально	1
реалізацію	1
реалізація	2

Лексична різноманітність: 55.97897503285151%

Fig. 11. The result of lexical analysis of news article

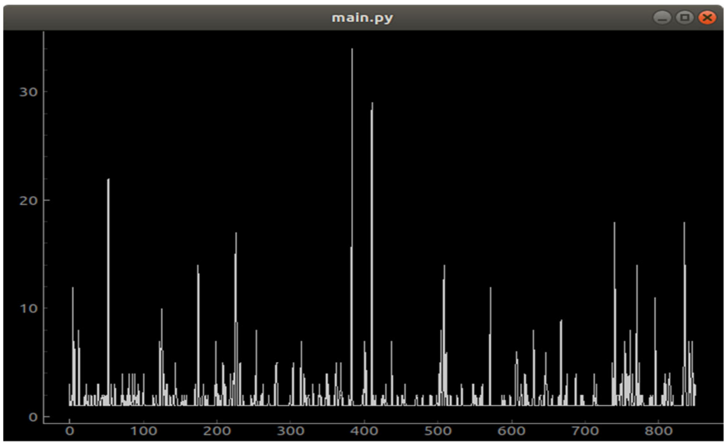


Fig. 12. News article

6 Conclusions

During the research, several natural language processing tools have been identified and reviewed. The advantages of these systems are that they perform natural language processing at several levels: text preprocessing, morphological, syntactic and semantic. The disadvantage of the systems is that not all of them provide the means of text processing in Ukrainian language at several levels of analysis, and the systems that have tools for processing Ukrainian texts produce incorrect results, in particular at the text preprocessing and morphological levels of the analysis.

As a result, the system that implements specific linguistic tasks related to the processing of Ukrainian language, namely, text preprocessing, morphological and lexical analyzes of text was developed. The basic stages of text preprocessing are considered in detail and algorithms of their program implementation were presented. In addition, the results of the developed system were demonstrated.

The advantage of the developed system is that it provides the features for complex processing of Ukrainian texts at three levels of analysis. The disadvantages of the developed system are: suboptimal algorithm of the assembly of words written with letter-spacing (the algorithm uses a dictionary to identify words written with letter-spacing; to reduce the volume of word search, the length limit is introduced, so words larger limit values could be convoluted with mistakes); incorrect analysis of non-vocabulary words, that is a disadvantage of pymorphy2, which is taken as the basis of the morphological module of the system; simple lexical analysis.

References

1. Indurkhya, N., Damerau, F.J.: Handbook of Natural Language Processing. 2nd edn. Chapman and Hall/CRC: Machine Learning & Pattern Recognition (2010).
2. Jurafsky, D., Martin, J.: Speech and Language Processing, 2nd edn. Upper Saddle River, N.J: Prentice Hall (2008).
3. Brown Corpus Manual, http://korpus.uib.no/icame/manuals/BROWN/INDEX.HTM., last accessed: 2020/02/14.
4. The Brown Corpus of Ukrainian Language, https://github.com/brown-uk/corpus, last accessed 2020/02/14.
5. Kübler, S., Zinsmeister, H., Corpus Linguistics and Linguistically Annotated Corpora. Bloomsbury Academic (2015).
6. Corpus of Ukrainian Language (Ukrainian), http://www.mova.info/corpus.aspx?l1=209, last accessed 2020/02/14.
7. Anthony, L.: A critical look at software tools in corpus linguistics. Linguistic Research 30 (2), 141–161 (2013).
8. Russian stemming algorithm, http://snowball.tartarus.org/algorithms/russian/stemmer.html, last accessed 2020/02/14.
9. Natural Language Toolkit - NLTK 3.5b1 documentation, www.nltk.org/book, last accessed: 2020/02/14.
10. Morphological analyzer pymorphy2 (Russian), https://pymorphy2.readthedocs.io/en/latest/, last accessed 2020/02/14.
11. brown-uk/dict_uk: Project to generate POS tag dictionary for Ukrainian language GitHub, https://github.com/brown-uk/dict_uk, last accessed 2020/02/14.
12. LT2OpenCorpora – GitHub, https://github.com/dchaplinsky/LT2OpenCorpora, last accessed 2020/02/14.
13. Rosette Text Analytics - AI for Human Language, https://www.rosette.com/, last accessed 2020/02/14.
14. Welcome to polyglot's documentation! - polyglot 16.07.04 documentation, https://polyglot.readthedocs.io/en/latest/, last accessed 2020/02/14.
15. Automatic text processing (Russian), http://www.aot.ru/, last accessed 2020/02/14.
16. Bird, S., Klein, E., and Loper, E.: Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit. O'Reilly Media, Inc. (2009)
17. Shortening of words in Ukrainian language in the bibliographic description. DSTU 3582-97 (Ukrainian), http://www.library.ukma.edu.ua/fileadmin/documents/Bibliography/26_DCTU3582-97.pdf, last accessed 2020/02/14.
18. GitHub - lang-uk/ner-uk: Ukranian NER annotation project, https://github.com/lang-uk/ner-uk, last accessed 2020/02/14.