International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 03 ISSUE 10 Pages: 190-199
SJIF IMPACT FACTOR (2021: 5.478) (2022: 5.636) (2023: 6.741)
OCLC – 1368736135

Journal Website: http://sciencebring.com/index.php/ijasr

Copyright: Original content from this work may be used under the terms of the creative commons attributes 4.0 licence.

Research Article

SOFTWARE OF THE NATIONAL CORPUS OF THE UZBEK LANGUAGE

Submission Date: October 10, 2023, **Accepted Date:** October 15, 2023,
Published Date: October 20, 2023
Crossref doi: https://doi.org/10.37547/ijasr-03-10-31

Tursunov Mukhammadsolikh
Samarkand Branch Of Tashkent University Of Information Technologies Named After Muhammad Al-Khwarezmi, Samarkand, Uzbekistan

ABSTRACT

This work is devoted to the creation of software for the national corpus of the Uzbek language. It describes the structure, components and tasks of the programs.

KEYWORDS

National corpus of the Uzbek language, software, model, algorithm, database, markup, token, concordance, search system.

INTRODUCTION

The study of natural languages using automated technologies based on reliable material is a promising area of modern science. One of the effective means of solving many linguistic issues is the electronic body of the language. The creation of such a system for the Uzbek language provides an opportunity to create new knowledge about the structure and lexical composition of the language, providing valuable material for the construction of linguistic models and the improvement of automated technologies for processing Uzbek texts.

Currently, the issue of creating a national Corpus of the Uzbek language is extremely relevant. Uzbek Corps linguistics is still at the initial stage of development. The real representative body of the Uzbek language has not yet been created. Scientific work on the creation of the linguistic supply of the Uzbek language Corps has been carried out [1,2] although the work on the

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 03 ISSUE 10 Pages: 190-199
SJIF IMPACT FACTOR (2021: 5.478) (2022: 5.636) (2023: 6.741)
OCLC – 1368736135

Crossref doi Google Scholar WorldCat MENDELEY

ISSN-2750-1396

software [3-6] is not yet sufficient. In this area, it is important to develop software products and establish their free use.

Software structure and tasks. The programs designed to create the national corpus of the Uzbek language consist of two parts (Fig. 1):

1. programs designed to create a corpus;
2. programs that serve to use the corpus.

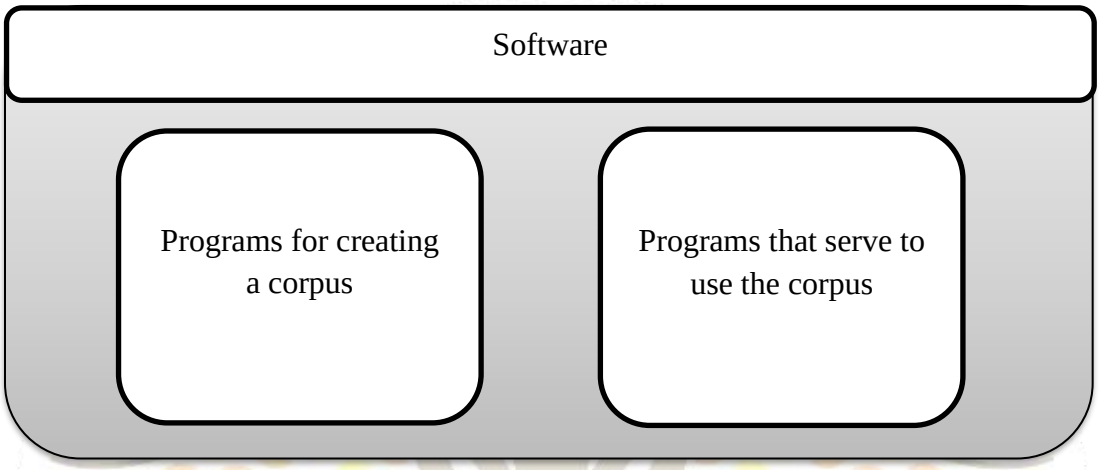

Figure 1. Software content

Programs designed to create a corpus should be able to perform tasks such as creating a text database, creating and editing a corpus dictionary, and text formatting (Fig. 2).

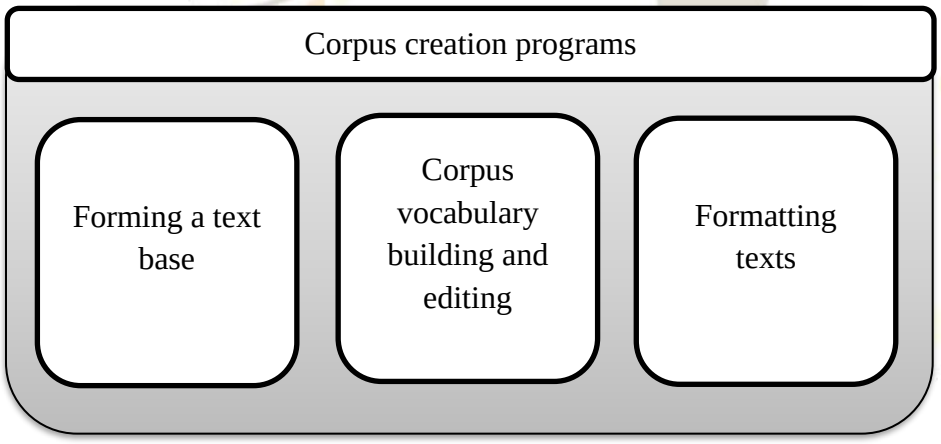

Figure 2. Components of corpus building programs

Text base creation programs. When creating a text base, the following tasks must be performed:

- digitization of texts, their editing;
- write text information to a file;
- entering the text into the database.

We select text digitization tools depending on the source of their initial state (paper, *.pdf format file).

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 03 ISSUE 10 Pages: 190-199
SJIF IMPACT FACTOR (2021: 5.478) (2022: 5.636) (2023: 6.741)
OCLC – 1368736135

Crossref doi Google Scholar WorldCat MENDELEY

ISSN-2750-1396

To digitize text from a paper source, we first scan it. Scanning hardware and scanning software are used for this. The resulting electronic text is converted to *.docx format using the Fine Reader program, then compared with the main source, it is carefully checked, errors are corrected and saved.

If the text is given in *.pdf format file, it will be converted to *.docx format using Fine Reader tools. It is then compared to the original source, scrutinized, corrected for errors, and saved.

The next stages of creating a text database are performed by a program named Dast_MtnBaza. This program forms the text base of the corpus based on the saved files. A text database consists of one folder named Matn_Base and one metadata file named MeteRazm in the computer's memory. *.docx files containing the texts decided to be included in the corpus are stored in the Matn_Baza folder. The MeteRazm file stores metadata information about the text contained in each file included in the Matn_Base folder. Metadata consists of general information about the text, including:

- text name;
- the time when the text was written;
- theme and type of text;
- genre;
- text size (in words).

The program ensures that these data are entered by the user and writes them to the MeteRazm file. Then, based on the entered metadata, a special unique name is formed, and the *.docx file in which the text is saved is copied to the Matn_Baza folder with this name.

Corpus vocabulary building and editing programs. For grammatical (morphological) classification of the text, the grammatical vocabulary of the language should serve as a basis. For example, A.A. Zaliznyak's grammar dictionary of the Russian language [7] serves as a basis for the national corpus of the Russian language. This dictionary is transferred to electronic form and is used for the classification of Russian words. But there is no such dictionary for the Uzbek language. Therefore, it is necessary to create a grammatical electronic dictionary of Uzbek words.

A program called Gram_Lugat was created to create a grammar dictionary. The dictionary is formed based on the texts included in the text base of the corpus. The Gram_Lugat program processes each file in the Text_Base text base of the corpus in turn. For each file, the Gram_Lugat program does the following:

• text tokenization;
• making different lists of words in the text (alphabetical-frequency, frequency-alphabetical and reverse list);
• entering words into the grammatical dictionary;
• browse and edit the grammar dictionary.

Text tokenization. In automatic text processing, first of all, the issue of extracting words from it, or in other words, dividing the text into units, arises. For this, all partial lines that do not contain separators (spaces, punctuation marks, etc.) should be separated from the text. And this will be a set of tokens [8]. One of the fundamental algorithms of automatic text processing consists in dividing the given text into tokens. The algorithm is given a text as input, and the output

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 03 ISSUE 10 Pages: 190-199
SJIF IMPACT FACTOR (2021: 5.478) (2022: 5.636) (2023: 6.741)
OCLC – 1368736135

is a list of tokens in the text. The program that implements this algorithm is called a tokenizer. Usually, tokens have the same meaning as word forms. However, to represent lexical units, the term "token" is used, not "word". This is because in some cases units smaller than a word (individual morpheme) or units larger than a word (word combinations) can be used as tokens [8].

The tokenizer breaks up the text, first, on the basis of the probes (space characters) between words, and then removes the punctuation marks from the words. Abbreviations (e.g. TATU (TITU), BMT (UNO), MDH (CIS), etc.) and date (e.g. 09.04.2018) are also taken as tokens [3].

The results of the tokenizer operation will be as follows:

Table 1. Tokenizer job result

№	Given text	List of tokens
1	*O‘zbekiston Respublikasi 02.03.1992 kuni BMT ga a’zo bo‘lgan*	*O‘zbekiston Respublikasi 02.03.1992 kuni BMT a’zo bo‘lgan*
2	*TATU Samarqand filiali 2005 yilda o‘z faoliyatini boshladi*	*TATU Samarqand filiali 2005 yilda o‘z faoliyatini boshladi*

Lexical decomposition is fundamental to automatic text analysis, as it serves as the basis for a number of other algorithms.

Making different lists of words in the text. The goal here is to extract text units (words, word forms) that should be included in the dictionary, make a list of them, and present the list in different forms (alphabetic-frequency, frequency-alphabetic, and reverse list) at the request of the user. Gram_Lugat program module suitable for this task is called Suz_Rhati, its input is a list of tokens separated from the text, and its output is an ordered list of words and word forms. To generate this list, the program uses the grammar dictionary file LUGAT. Each token given on input is looked up in an existing LUGAT file. If a token is found in the dictionary, then it does not need to be included in the dictionary, it can be discarded. Otherwise, i.e. if the token is not found in the dictionary, it is checked whether it is a word

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 03 ISSUE 10 Pages: 190-199
SJIF IMPACT FACTOR (2021: 5.478) (2022: 5.636) (2023: 6.741)
OCLC – 1368736135

Crossref doi Google Scholar WorldCat MENDELEY

ISSN-2750-1396

or word form. Tokens that are words or word forms are listed separately. This list is sorted and displayed according to the user's request, either in alphabetical-frequency order, or in frequency-alphabetical order, or in the form of a reverse list. Entering words to the grammar dictionary. The software module performing this task is called Lug_Kirit. Adding a new unit to the dictionary is done by the user in interactive mode. This process is provided by the Lug_Kirit program. On the computer screen there is a list issued by the Suz_Rhati program, and one word is highlighted in it with a different color. Using the Lug_Kirit program, the user moves through the list, selects the current word and presses the "Enter" button. As a result, an input window appears on the screen (Fig. 5). The user enters the requested data in this window from the keyboard and clicks the "Input to database" button. Then the selected word and the corresponding grammatical characteristics are summarized and recorded in the electronic dictionary file LUGAT and other auxiliary files. After that, the user selects the next word from the list and enters it into the database. In this way, all the words in the given list are included in the LUGAT file, and the grammatical dictionary is enriched step by step.

Text formatting programs. Grammarly parser is built on Nuxt JS, Python and PostGreSql database management system (DMS). In order to simplify the process of entering texts into the corpus, grammaticalization is performed in two stages: the stage of initial formatting of the text and the stage of parsing. At the initial formatting stage, the structural components of the text are determined, that is, words, sentences, paragraphs and paragraphs are separated in the text.

Initial formatting step. At this stage, regardless of the original format of the file to be included in the corpus, it must be converted to MS Word format. It is necessary to use MS Word versions 2007 and higher, because in them the text is in *.docx format and is automatically tagged based on the standard system. Dividing the text into structural components (words, sentences, paragraphs, paragraphs) is performed automatically by the MS Word program. Information indicating the position of words in the text is written to a separate file.

Layout stage. Parsing should be understood as attaching special tags to the text and its components. Special tags are of two types: linguistic tags and extralinguistic (external) tags. Linguistic tags consist of information that describes the lexical, grammatical, and other similar properties of text elements. And extralinguistic tags describe information about the author and the text (author, title, year and place of publication, genre, subject matter, etc.) [9].

In proportion to the huge size of the hull, the layout is a time-consuming and labor-intensive process. Therefore, it should be done automatically. If some type of sorting has not been automated yet, then it will have to be done manually.

At this stage, initially, the automatic layout program is launched. As a result, the text is presented in the form of a *.docx format file (Fig. 4), in which the coded words are distinguished by red and the uncoded words by black. (If the

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 03 ISSUE 10 Pages: 190-199
SJIF IMPACT FACTOR (2021: 5.478) (2022: 5.636) (2023: 6.741)
OCLC – 1368736135

grammar dictionary is empty, all words are displayed in black.)

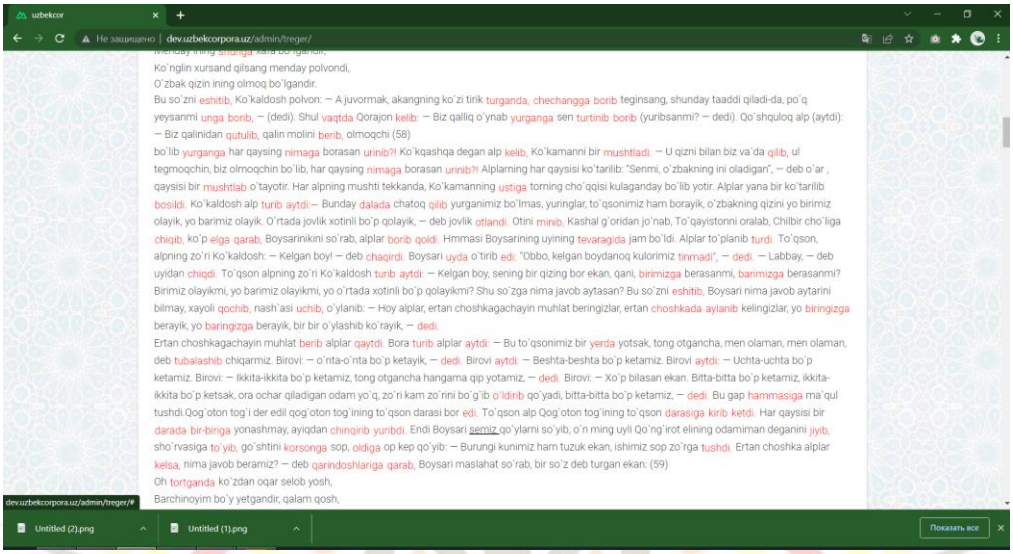

Figure 4. The initial state of the file

For this, when the cursor is pressed on the word, a window with morphological characteristics of the word appears on the screen (Fig. 5). From the information in this window.

a) the word is not sorted **b) the case where the word is arranged**

Figure 5. Grammatical tags of the word

Using the colors of the words in the text, it is determined whether the word is tagged or not. If the word has not yet been classified, the corresponding button is clicked and a window for specifying the morphological parameters of the word is opened (Fig. 6).

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 03 ISSUE 10 Pages: 190-199
SJIF IMPACT FACTOR (2021: 5.478) (2022: 5.636) (2023: 6.741)
OCLC – 1368736135

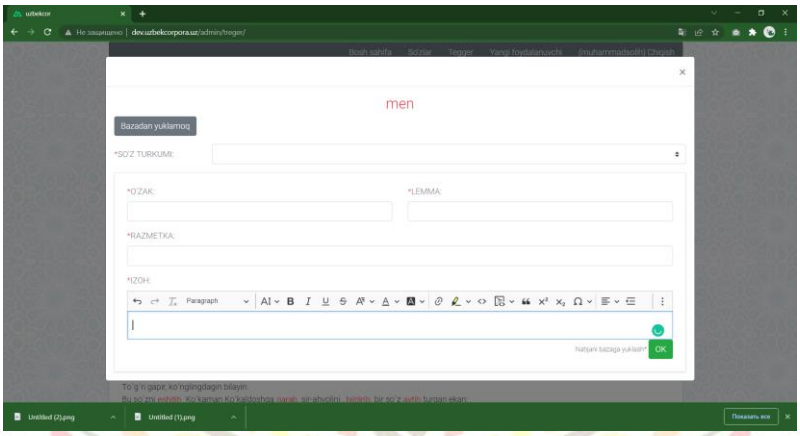

Figure 6. A window for assigning morphological characteristics to a word

Figure 6 shows the morphological classification of the word. In this case, its category, grammatical symbols, stem, lemma and explanation are included in the word. When we select a category from the window, a list of grammatical characters corresponding to the category opens. The desired characters are selected from the list.

Tegger software is used to insert texts into the corpus. In the process of adding texts to the corpus, the text is processed grammatically. Grammatical information is filled into words in the text semi-automatically, and words with grammatical information change color to red, and the user can clearly distinguish between unmarked and marked words. Linguistic support for word classification is provided in the appendix. This can be seen in Figure 4. To assign grammatical information to unclassified words, click once on the word and a window for grammatical processing of the word will appear (Figure 6). After the word is sorted, the color of the word will change to blue. This means that when working with the text, the format is entered manually. When you click on a word in red, its grammatical information is displayed. If the grammatical information is given incorrectly, the "Change" button is clicked from the window in b of Fig. 5, and the window in Fig. 6 is created for grammatical processing of the word. After processing the text, the text is saved again. Texts are stored in the corpus in JSON format.

Programs that serve to use the corpus. The software should enable the user to conduct linguistic research on the texts contained in the corpus and draw conclusions based on this. In particular, the following tasks are assigned to the software by the corpus user:

- Creating a concordance;
- Search for contexts not only by words, but also by phrases;
- Sort lists by several criteria selected by the user;
- Provide an opportunity to describe the found word forms in an expanded context;
- Providing statistical information on separate elements of the corps;
- Save and print results;
- Ability to work not only with individual files, but also with unlimited size cases;

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 03 ISSUE 10 Pages: 190-199
SJIF IMPACT FACTOR (2021: 5.478) (2022: 5.636) (2023: 6.741)
OCLC – 1368736135

• Quick response to inquiries and quick release of results.

In general, the performance of these tasks consists of processes such as finding the necessary information for research, collecting it and presenting it to the user in the right way. In order to implement these processes, the issue of information retrieval across the corpus must be solved.

The software can perform several types of search in the entire corpus or a selected part of it: search by specific form (the user enters the form of the word being searched for); search by initial form (the user enters the initial form of the word and gives a request to extract all other forms); search by grammatical characters. In the search based on grammatical characters, the user is presented with a list of all the parameters used in the grammatical analysis, and he selects and defines the parameters he is interested in, and after that, the search is performed based on the specified parameters [10].

When solving the search problem, of course, the problem of reducing the time of information processing arises. The effectiveness of solving this problem depends on which database management system is used. We use PostGreSQL DMS. The reason for its selection is that it is currently free and compatible with many software systems. In addition, in order to solve the problem of increasing the speed of information processing and presentation, it is possible to support, compare and select different structures of information.

A special function is invoked when a request is received to extract the context of a word. This function is passed as a parameter a value indicating the position of the key word in the context in the specific text. Then, to each word in the text, its information in the database is attached as a marker. An empty marker is attached to words that are not found in the database. After that, the context is displayed on the computer screen. The search word is highlighted in contexts and highlighted in red. This scheme of presenting contexts is also applied to words that are not grammatically analyzed. A program called a special parser breaks the text into structural units, and by searching for words with empty markers, ungrammatical words are found. Punctuation can also be found by adjusting the settings of the defragmenter.

A simplified scheme of the program is shown in Figure 7.

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 03 ISSUE 10 Pages: 190-199
SJIF IMPACT FACTOR (2021: 5.478) (2022: 5.636) (2023: 6.741)
OCLC – 1368736135

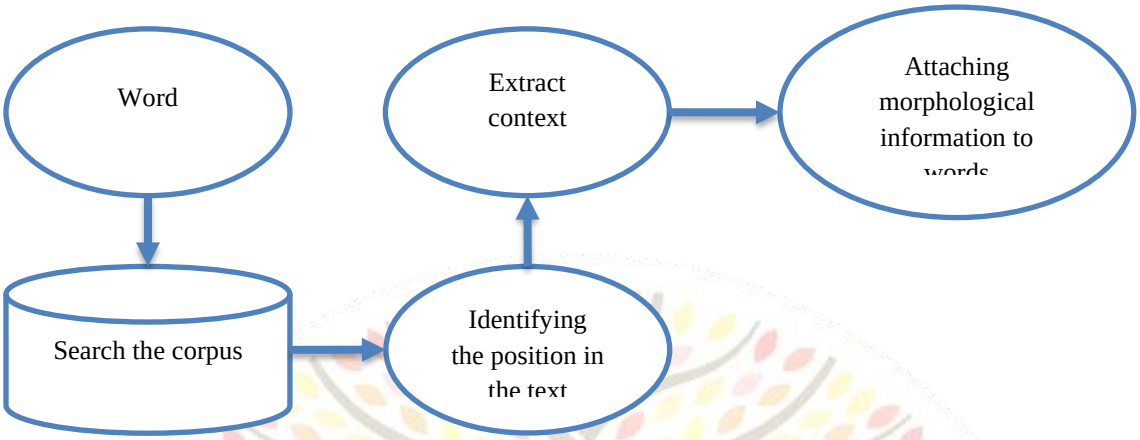

Figure 7. Scheme for searching words in the corpus.

In the first case, grammatical characteristics were extracted only for the search query in the context. Later, with the advice of philological experts, the grammatical characteristics of all the words in the context were presented (Figure 8). At the same time, the ability to switch to an expanded context was also created.

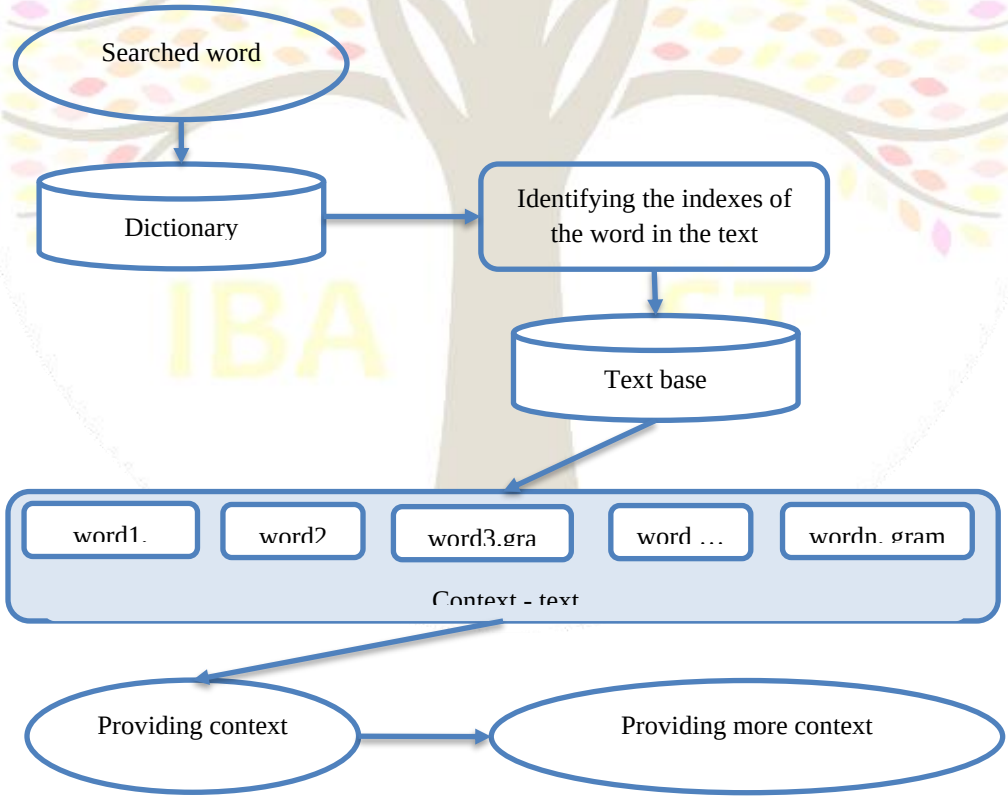

Figure 8. Corpus word search and presentation of contexts

International Journal of Advance Scientific Research
(ISSN – 2750-1396)
VOLUME 03 ISSUE 10 Pages: 190-199
SJIF IMPACT FACTOR (2021: 5.478) (2022: 5.636) (2023: 6.741)
OCLC – 1368736135

The software performs word searches across the corpus in several forms.

Conclusion

The Uzbek language text corpus software consists of two parts, it should include programs designed to create the corpus and programs that serve to use the corpus.

Programs designed to create a corpus perform tasks such as creating a text base, creating and editing a corpus dictionary, and text formatting.

For the grammatical (morphological) classification of the text, the grammatical dictionary of the Uzbek language should serve as the basis. But there is no such dictionary for the Uzbek language. Therefore, it is necessary to create a grammatical electronic dictionary of Uzbek words.

To work with the corpus, it is necessary to create web resources that allow not only local use, but also Internet access. It works well to have options for grammar and syntax analysis and error correction done online.

References

1. Mengliyev B. National corpus of the Uzbek language // Enlightenment. - 26.04.2018
2. Khamroeva Sh.M. Linguistic support of the morphological analyzer of the Uzbek language, Monograph. – Tashkent, 2020.
3. Tursunov M.S., Karshiev A.B., Karimov S.A., Development of a modern corpus of computational linguistics. Scopus, DOI: 10.1109/ICISCT50599.2020.9351376, noyabr 2020.
4. Karshiev A.B., Tursunov M.S., Kholmukhamedov B. Uzbekcorpora.uz: Uzbek language corpus software. Native languages and cultures in today's changing world. Electronic network scientific journal. No. 1, 2022, - pp. 71-78.
5. Tursunov M.S., Karshiev A.B. Programs for creating an electronic dictionary of the Uzbek language, Software certificate, DGU 13733, Uzbekistan, 2021.
6. Tursunov M.S., Karshiev A.B., Kudratov I.L. Database management system of the national corpus of the Uzbek language, Software certificate, DGU 14440, Uzbekistan, 2021.
7. А.А.Зализняк, грамматический словар русского языка. Москва, 1980.
8. Николаев И.С., Митренина О.В., Ландо Т.М. (ред.) Прикладная и компьютерная лингвистика. Ленанд, 2016. - 320 с. - ISBN 978-5-9710-3472-8.
9. Бочаров, В.В. Программное обеспечение для коллективной работы над морфологической разметкой корпуса / В.В.Бочаров, Д.В.Грановский // Труды международной конференции «Корпуснаялингвистика-2011». -СПб.:С. – Петербургский государственный университет, 2011.
10. Tursunov M.S., Karshiyev A.B., Karimov S.A. Preliminary results of the creation of software for the Uzbek language corpus. Scientific-practical and information-analytical journal of the descendants of Muhammad al-Khorazmi, 1(15), March 2021.