

# IT Slang Analysis System

Hanna Kravchenko, Zoriana Rybchak, Olha Kulyna, Tetiana Shestakevych

Lviv Polytechnic National University, S. Bandery street, Lviv, 79013, Ukraine

## Abstract

The present study experimentally has investigated IT slang in the Ukrainian language. It is used by both regular users and specialists who work with information technologies. Professional slang used by IT experts in Ukraine is based on common programming vocabulary that is originated in the English language. The majority of slang terms used in the IT industry refer to certain specialty ideas and procedures. Our aim with this paper is to develop a chatbot which helps to search and define slang words used in the IT industry. The implementation was carried out using NodeJS, MongoDB and SpaCy. The program consists of a convenient interface and a search engine that allows users to find definitions of words with a set of relevant processes. A chatbot is integrated in Telegram which is one of the most well-known instant messaging platform and has high cross-platform capability.

## Keywords

IT slang, NodeJS, MongoDB, SpaCy, database, Natural Language Processing, Telegram.

## 1. Introduction

Slang is a means of marking the speaker's affiliation to a relevant social group or circle of people who share their interests, activities and hobbies. There is a sufficiently diverse and specific language, in the computer culture which is called "IT slang" or "computer slang" [1; 2; 3;4]. IT slang in the Ukrainian language is formed based on English words; it is used by both ordinary users and professionals who deal with information technologies. Common programming vocabulary borrowed from the English language, is the basis of professional slang shared by IT specialists [5; 6; 8; 9]. Most slang expressions in the IT field are narrow niche concepts and processes.

The aim of this research is to develop an IT slang analysis system that will allow users to get an explanation of a word used by people in the IT sphere. In order to achieve the set goal, the following tasks must be solved:

- to carry out an analysis of literary sources, on the basis of which to investigate the current IT slang establishment and development;
- to analyze and compare existing systems;
- to conduct a systematic review of the system, in particular, create a goal tree and UML diagrams that will emphasize and demonstrate the functionality;
- to choose the appropriate methods and means for implementing the set task using the analyzed data, which will be most suitable for solving the problem;
- to develop a software based on the created and described functionality of the system.

The adoption of a chatbot has been hailed as a revolution in how people interact with technology and businesses. Compared to traditional apps, they feature a more straightforward user experience and chatbot should be able to comprehend and carry out any requests made by users [11; 19; 20].

MoMLeT+DS 2023: 5th International Workshop on Modern Machine Learning Technologies and Data Science, June 3, 2023, Lviv, Ukraine

EMAIL: hanna.v.kravchenko@lpnu.ua (H. Kravchenko); zoriana.l.rybchak@lpnu.ua (Z. Rybchak); olha.v.kulyna@lpnu.ua (O. Kulyna); tetiana.v.shestakevych@lpnu.ua (T. Shestakevych)

ORCID: 0009-0005-0271-5442 (H. Kravchenko); 0000-0002-5986-4618 (Z. Rybchak); 0000-0002-2334-0660 (O. Kulyna); 0000-0002-4898-6927 (T. Shestakevych)



© 2023 Copyright for this paper by its authors.  
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).  
CEUR Workshop Proceedings (CEUR-WS.org)

Many companies are switching to chatbot systems for customer assistance. The reason for that lies in the enormous cost reduction compared to real people.

Our Chatbot has been integrated in Telegram. This instant messaging platform has a great number of advantages and can be used on Android, iOS, Windows and many other platforms that can support the web version.

## 2. Related works

The development of computer terminology caused another natural process from the linguistic perspectives - the formation of computer slang. There are different opinions about the emergence of computer slang: some linguists believe that this type of language arose simultaneously with the appearance of electronic computing machines in the USA in 1946, others are of the opinion that computer slang as a separate type of slang "appeared only in the 60s of the XX century and associate this process with the so-called "minicomputer era"[1; 6; 7]. On the one hand the emergence of slangisms is a negative phenomenon, because it "clutters" the literary language, violates the norms, makes it incomprehensible, and coarse. On the other hand, the existence of computer slang allows specialists not only to feel like members of a closed community, but also helps them understand each other better.

There is a number of research devoted to chatbots. Ch. V. Misischia, F. Poecze, Ch. Strauss discuss the chatbots in customer service, their relevance and impact on service quality. They agree that Chatbots are quickly making their way into e-commerce and e-services as their adoption offers significant prospects to enhance customer support. In this study, chatbots are investigated in this context with a focus on the functional elements that are already noticeably raising service quality. Based on a review of recent publications in the field, an overview of their key features and functionalities is first provided, highlighting the value of chatbots for customer support. Second, a new contribution is made by presenting two different chatbot goals based on their functional dedication: "improving service performance" and "fulfilling customer expectations." Among the customer-related activities of chatbots that are taken into consideration are interaction, amusement, problem-solving, trendiness, and customization. The categories of chatbots are covered in detail [14].

Another study by Caldarini G, Jaf S. And McGarry K. emphasize on importance of Intelligent conversational computer programs known as chatbots. They are created to mimic human speech in order to provide automated online support and guidance. In order to give clients virtual support, chatbots have been widely used by numerous businesses due to their growing benefits. Natural Language Processing and Machine Learning are two areas of artificial intelligence that are used by chatbots. However, there are numerous obstacles and restrictions in their use. This survey looks at current developments in artificial intelligence and natural language processing applied to chatbots. It emphasizes the key difficulties and shortcomings of the current research and offers suggestions for more study [15].

The development of MongoDB was contributed by Kudo T., Ishino M., Saotome K., and Kataoka W. who suggest solution for dealing with a large amount of data in database and propose a method to process plural data as a single transection for MongoDB [16]. Chauhan D. And Bansal K. provide overview of NOSQL database which increases data access and storage efficiency and scalability. The authors conduct a case study and propose a method to process plural data as a single transection for Mongo DB [17].

Under this heading we also would like to analyze two Ukrainian existing systems which perform language processing. The first one, Slovotvir (<https://slovotvir.org.ua>) is a platform for translation of borrowed words. Users benefit from searching, discussing and matching for borrowed words. The system has ongoing updates, and everyone can contribute to it. It has some drawbacks as well, namely it is not integrated in social networks, interaction with third-party application is limited and there is no section dedicated to computer slang.

Another project Wiktionary (<https://uk.wiktionary.org>) is a free multilingual dictionary which has a Ukrainian section and free-to-use contribution. It contains explanations and translations of Ukrainian words as well as translations of words and phrases from other languages. There are three

features of the systems which are highly valued: contains a huge number of words and has a vast number of users; provides in-depth analysis of the semantic properties of words; a free contribution to the project. Yet the support is limited, interface is rather outdated, and no ARI is available for third-party integration.

A significant number of scientists contributed to the development of chatbot technology which uses artificial intelligence and natural language processing to respond to customer questions and simulate human conversations [10; 12; 14; 19; 20]. These days chatbot gain their popularity providing responses without the need for human intervention.

### 3. Methods and means of task solving

The system provides an opportunity to perform a search by entering a word and searching for a given word in the dictionary to output a translation (Ukrainian-English), as well as receiving a definition of the word. The system provides an opportunity to search by parts of speech, namely by such parts as Noun, Adjective and Verb. In order to use the functionality, the user must enter the bot and select the section required for work. The information system helps the corresponding user independently to carry out training with the help of words of interest or parts of speech.

The following main effects should be noted after the implementation and start-up of the system: 1) **Economic** - indicates the necessity and expediency of attracting financial contributions to the project. 2) **Financial** - shows the final profit from the implementation of the system. 3) **Time-frame** - shows the time frame for the development and implementation of the system and the time when it can be completed more quickly. 4) **Technological** - shows an increase in productivity and quality of working conditions. 5) **Scientific** - shows the novelty of the implemented project. 6) **Social** - shows the impact of the implemented project on people's lives. To assess the feasibility of implementing the system, it is necessary to explain the purpose, the effect, the unit of measurement and the estimated value of the effect. Effect scale can vary as very low, low, medium, high, very high. The value scale can be as following insignificant, secondary, important, very important, critical. The table 1 shows the assessment of the skills implementing IT projects and analyzing the effects of system implementation.

**Table 1**  
Assessment of the system implementation

| aim        | effect                   | unit of mesearment | assessment value |
|------------|--------------------------|--------------------|------------------|
| scientific | educational              | high               | important        |
| technical  | scientific and technical | very high          | essential        |
| economic   | economic and finance     | average            | secondary        |
| social     | time-frame and social    | average            | essential        |

The table defines the following goals: •1) ensuring the development of IT slang analysis algorithms; 2) creation of a platform for IT slang analysis; 3) the profit received from the launch of the system in accordance with the invested funds in development and implementation; 4) ensuring the required number of users, providing them with quality services for searching and interpreting IT slang.

The following algorithms were used for effective work and recognition of words entered by the user when designing the IT slang analysis information system: Fuzzy matching (fuzzy line search) and SpaCy mechanisms and components. Fuzzy matching is a technique used in computer translation as a special case of matching records. It works with matches that may be less than 100% perfect when searching for matches between segments of text and records in the previous translation database. It usually works on segments at the sentence level, but some translation technologies allow matching at

the phrase level. It is used when the translator works with a translation memory (TM). Another feature is usage of approximate string matching.

The closeness of the match is measured by the number of primitive operations required to transform the tape into an exact match. This quantity is called the editorial distance between the tape and the pattern. A transposition operation, in which two letters in a string are swapped, is also sometimes considered to be a primitive operation. The change  $abc \rightarrow acb$  is a transposition. Different approximate calculators impose different restrictions. Some calculators use a single global unweighted cost, that is, the total number of primitive operations required to transform a pattern match. For example, if the pattern is coil, foil differs by one substitution, oil by one deletion, and foal by two substitutions. If you consider that a single operation is worth one cost, and set a limit for one cost, then coil, foil and oil will be valid matches, and foal - not. Other calculators specify the number of operations of each type separately, while all others set a total value but allow different weights to be assigned to different operations. Some calculators allow separate setting of limits and weights for individual groups in the structure.

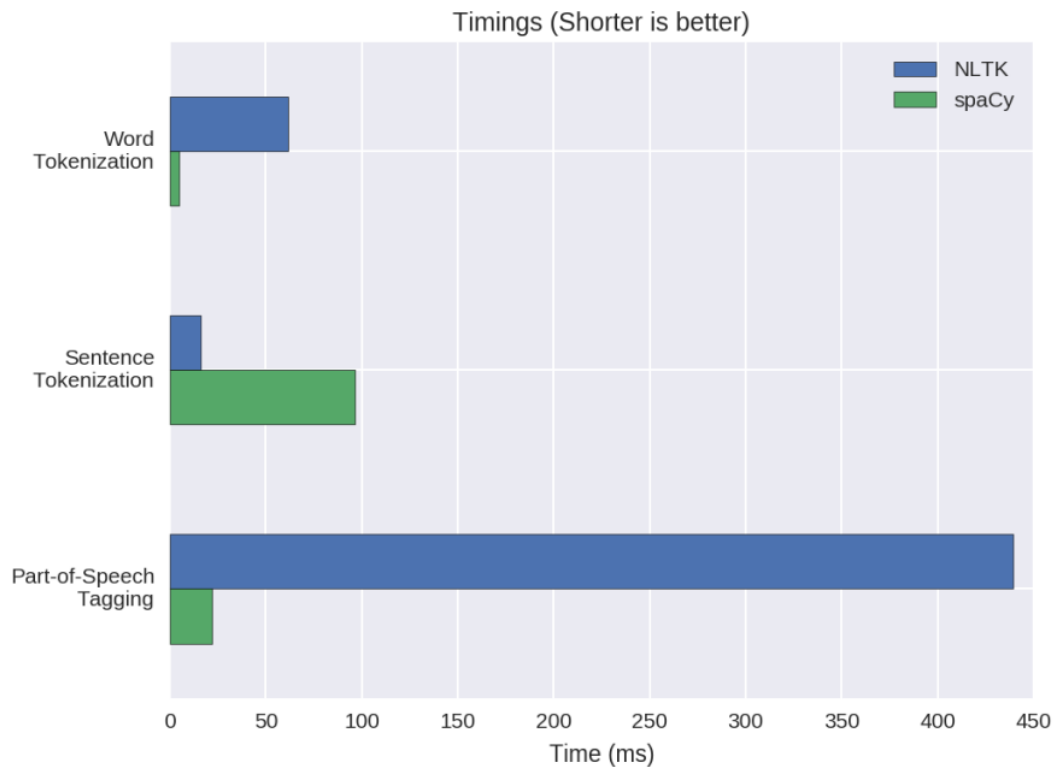
Let us consider a possible definition of the problem of approximate string matching. Pattern tape  $P = p_1p_2\dots p_m$  and a text tape  $T = t_1t_2\dots t_n$ , are known. A tape  $T_j = t_j\dots t_j$  in  $T$ , which of all sub-tapes  $T$  has the smallest editorial distance to pattern  $P$  and is unknown. A brute-force approach would be to calculate the editorial distance to  $P$  for all substraces of  $T$ , and then choose the subtrace with the minimum distance. But this algorithm would have a complexity of  $O(n^3 m)$ . A better approach is based on dynamic programming. It uses an alternative formulation of the problem: for each position  $j$  in the text  $T$ , as well as each position  $i$  in the pattern  $P$ , calculate the minimum editorial distance between the  $i$  first letters of the pattern,  $P_i$ , and any substring  $T_j = t_j\dots t_j$  in  $T$  ending at position  $j$ . For each position  $j$  in the text  $T$ , and each position  $i$  in the pattern  $P$ , go through all substrings of  $T$  ending at position  $j$  and determine which of them has the minimum editorial distance to the  $i$  first letters of the pattern  $P$ .

Computing  $E(m, j)$  is very similar to computing the editorial distance between two tapes. In fact, we can use Lowenstein's algorithm for  $E(m, j)$ , the difference is that we must initialize the first row with zeros, and keep the order of calculation, as in  $E(i - 1, j)$ ,  $E(i, j - 1)$  or  $E(i - 1, j - 1)$  in the calculation of  $E(i, j)$ . In the array that stores  $E(x, y)$  values, we then select the smallest value in the last row, for example  $E(x_2, y_2)$ , and continue the calculation in the other direction, to row number 0. If the field we reached is  $E(0, y_1)$ , then  $T[y_1 + 1] \dots T[y_2]$  is a substring of  $T$  with the minimum editorial distance to the pattern  $P$ . Calculating  $E(x, y)$  array takes  $O(mn)$  time with the dynamic programming algorithm, while the reverse calculation takes  $O(n + m)$  time. Until recently, approximate search was most often used in spell-checking systems. Also, due to the availability of large volumes of DNA data, comparison of nucleotide sequences has become an important task. Approximate comparisons are used in spam filters. But approximate comparison cannot be used for binary data such as images or music. They require other algorithms, such as "acoustic footprints".

The following are spaCy's rule-based mechanisms and components that not only allow you to find the right words and phrases – they also provide access to the markers in the document and their relationships. This means it's easy to access and parse surrounding tokens, merge ranges into individual tokens, or add entries to named entities in `doc.ents`. Natural Language Processing (NLP) is becoming very popular today because it is undoubtedly easier for people to communicate with machines in the same way as they communicate with people. Now, along with the rapid development of this area, more and more services use NLP: chatbots in which you no longer need to choose ready-made answers, voice assistants, e-mail to automatically sort letters, etc.

NLTK (Natural Language ToolKit) is the most famous NLP library created by researchers in this field. It is popular in academic circles and is mainly used for learning or creating different processing methods and not only using the basic tools that NLTK provides in a huge number and not always the best. Also, it is quite slow due to the fact that it is written entirely in Python and works with strings. SpaCy is in some ways the opposite of NLTK. It is much faster because it is written in Cython and works with objects. SpaCy basically provides the best tools for a specific task. In general, SpaCy with its models, speed, convenient API and abstraction is much better suited for developers creating ready-made solutions, and NLTK with a huge number of tools and the ability to garden any gardens - for researchers and students. In no case, neither one nor the other library is suitable for creating your own

models. Tensorflow, PyTorch and others are on offer. In short, SpaCy can do about the same as NLTK and their counterparts, but faster and more accurate.



**Figure 1:** Performance comparison between SpaCy and NLTK

The central data in SpaCy are Doc and Vocab, they are shown in Fig. 2. Doc stores sequences of tokens and their annotations. Vocab stores a set of reference tables that make common information available to all documents. There is no need to store multiple copies of this data with centralized storage of strings, word vectors and lexical attributes. This saves memory and provides a single source of truth. Doc holds the data, and the Span and Token are representations, allowing Spacy to run quickly without redundant copies pointing to them [13]. Doc is created by the Tokenizer, then is modified in-place by pipeline components. Language coordinates these components. It takes the raw text and sends it to the pipeline, returning an annotated document.

This library is good for both developers and researchers. SpaCy hides unnecessary details, allowing you to quickly solve NLP tasks with state-of-the-art solutions [18]. At the same time, the ability to finish the models and customize the pipeline frees up hands for new experiments.

Node.JS is not the only programming environment that runs on JavaScript, it is certainly the most common. It is a free and open-source cross-platform runtime that runs on JavaScript. Designed primarily for the server or client side of a mobile application, it is a full-stack development environment that separates tasks into completely separate "nodes". Each task is split into independent, separate node paths, tasks can run concurrently and seamlessly together without threatening the server or consuming too much power. Node calls this a "microservice" pattern, each of which is a self-contained path to perform a specific service. This is an innovative way to break a program into its smallest pieces. But it's a very effective way to work with mobile applications that need speed, accessibility and accuracy above all else.

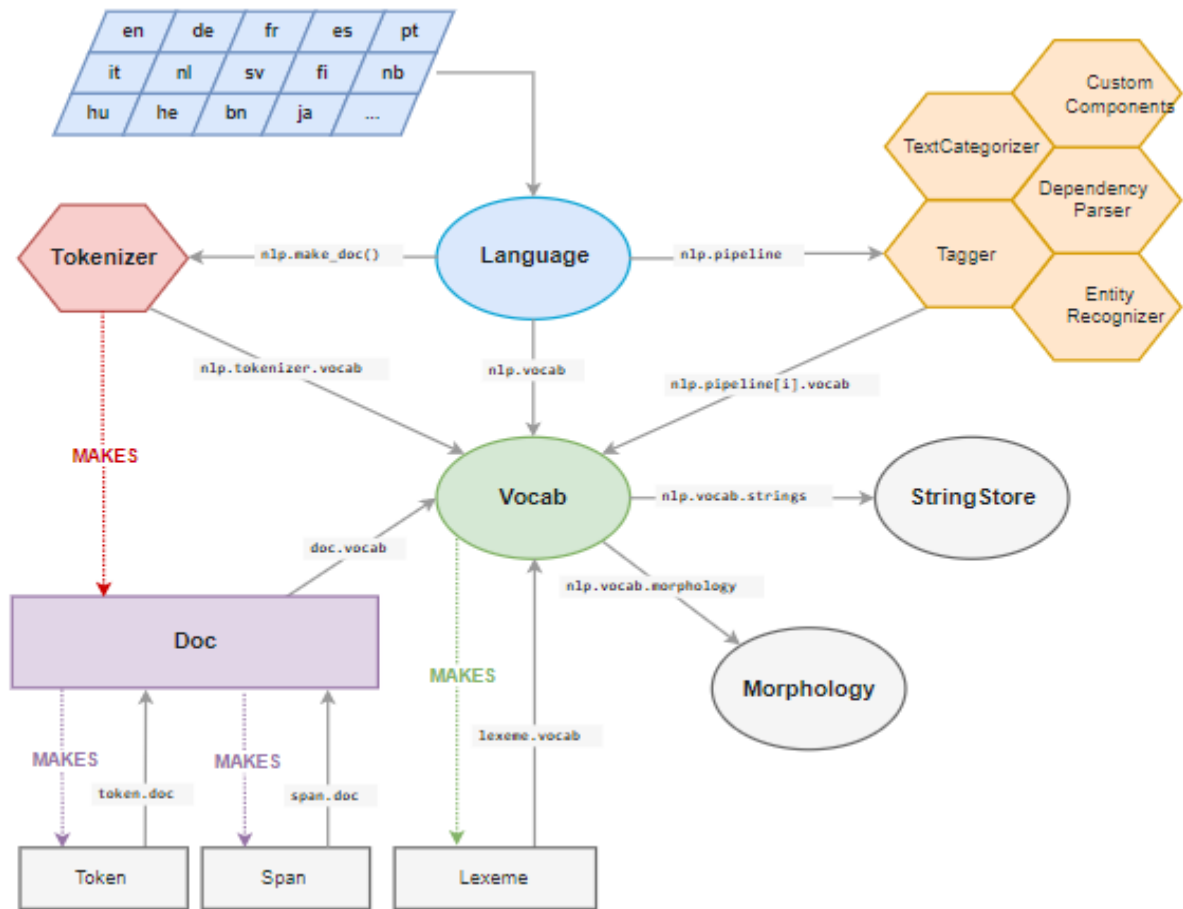


Figure 2: Structure of SpaCy database

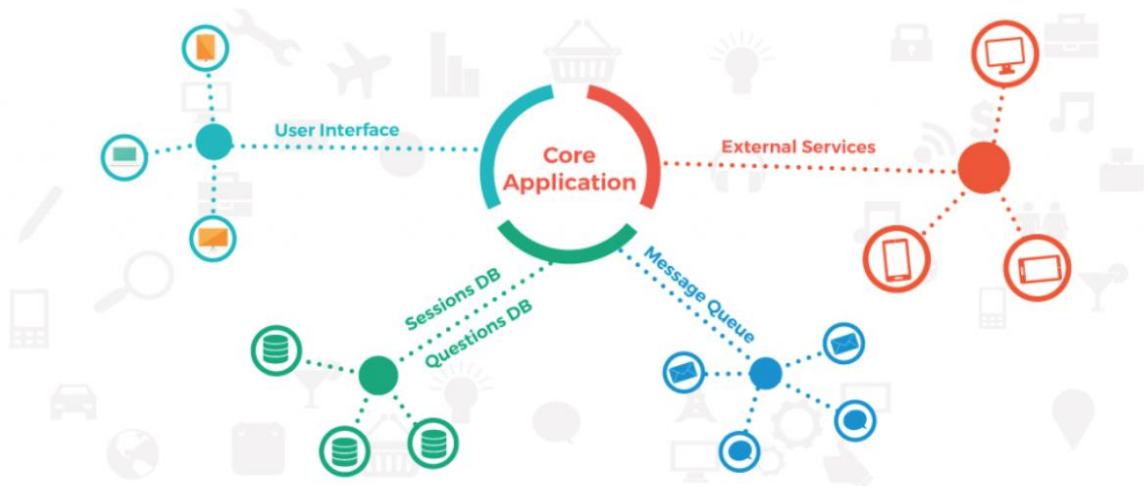


Figure 3. Microservice operation

The figure above shows how a typical program might be organized. Autonomous regions can work without interfering with each other or the main application itself. This structure has specific implications for the framework. With this capability, Node.js has the key advantage of being able to run in a single thread using non-blocking I/O calls. This allows it to handle tens of thousands of concurrent threads held in an event loop. It is important to note here that Node is not the only single-threaded platform. But it is one of the most common and popular.

NodeJS is a software tool that will be used to implement a software product. The main advantages are the speed and ease of development, as well as the possibility of scaling and maintaining the software product in the future. This is the platform that will be the basis for the implementation of the described program.

## 4. Results

A chatbot has been created to search and define words used in the IT industry (IT slang). The implementation has been carried out using NodeJS and MongoDB. The program consists of a convenient interface and a search engine that allows you to find word definition. The database stores data about the user's current steps to completion and guide the user on what steps they should take.

The database contains information about the user's actions. MongoDB was used as no other intermediate information will be stored during the design and development [13]. The view of the JSON structure used for the project contains the following fields, which are shown in Fig. 4 and Fig. 5:

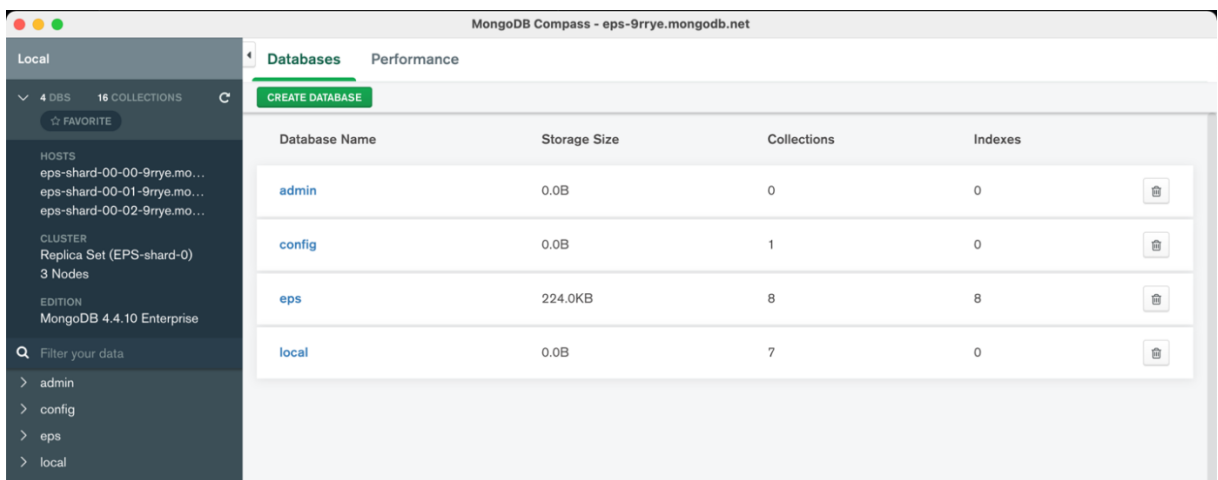


Figure 4. DB to deploy and work with MongoDB

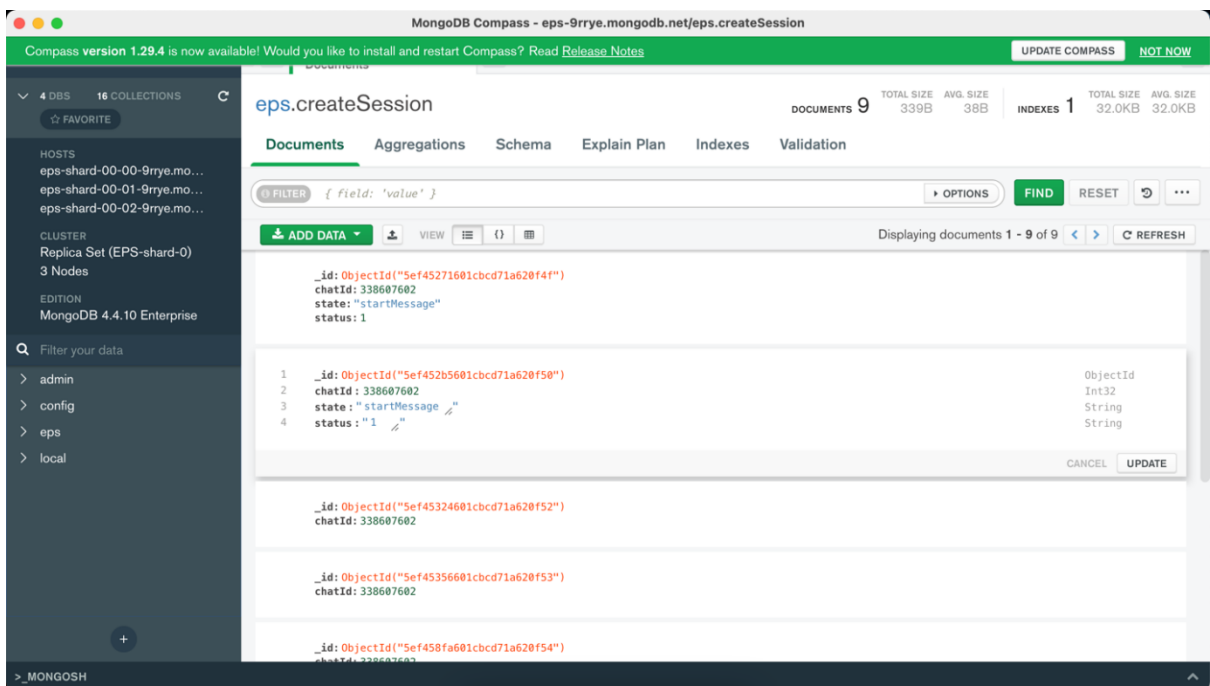


Figure 5. Data structure for storing user information

The aim of this program is to carry out word analysis in IT industry. The application is written using NodeJS. The main functions of the program:

1. search for the definition of the entered word;
2. sorting of the word by part of speech;
3. search for a word definition by selecting appropriate categories.

The system works as one user only application. The system can be used on any device with an installed application/web version of the Telegram messenger. The input data is the search word selected or entered by the user. The output data is information about the definition of a certain word.

Figure 6 presents depiction of a user's possible interaction with IT Slang Analysis Information System. It shows various use cases with only one user at a time.

The diagram shows the following use cases:

- Activate the bot /start.
- - Go to the main menu.
- - Select option "Search for word meaning".
- - Enter a word you search for.
- - Receive a reply (definition and translation).
- - Select "View list of categories".
- - Select a category.
- - Receive a response providing a definition and translation.

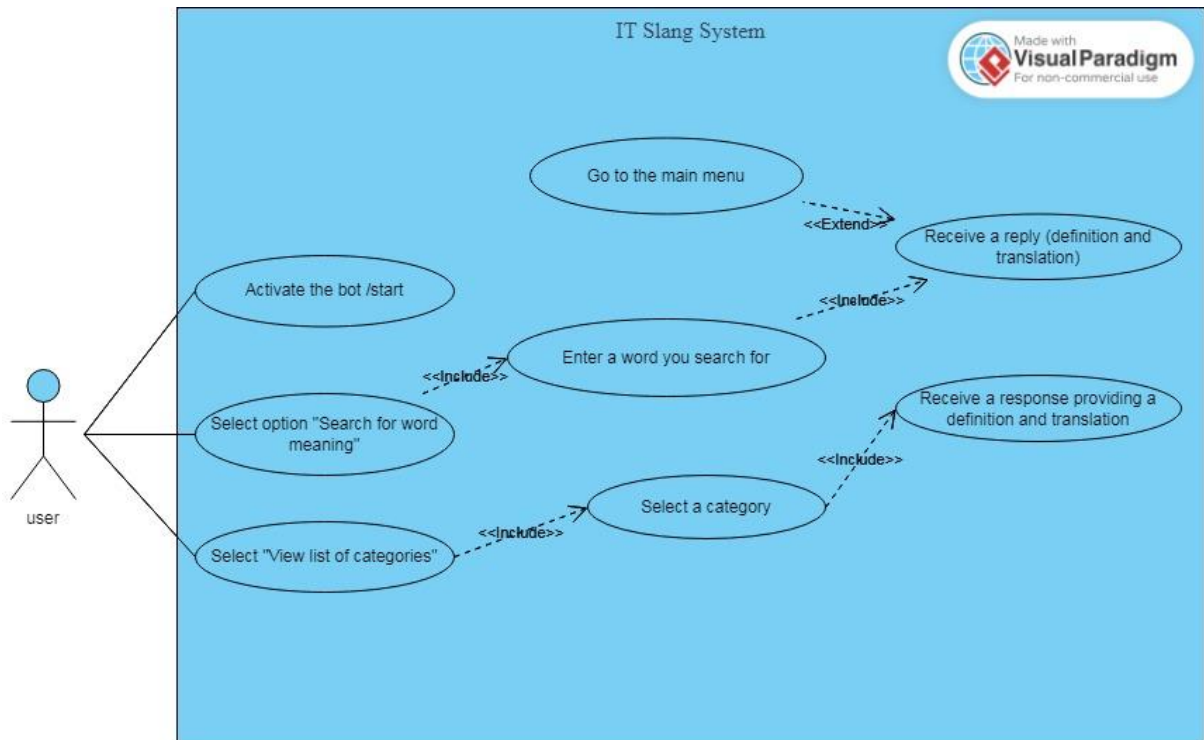
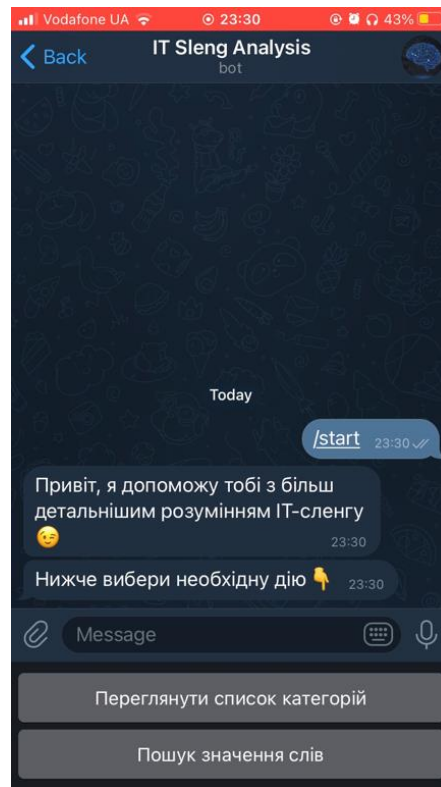


Figure 6. Use case diagram

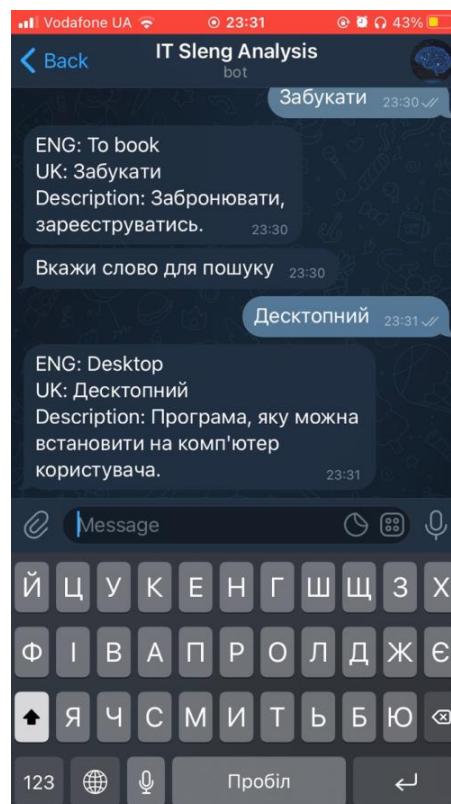
The next step is for the user to select the mode in which he wants to search by entering words or searching by parts of speech. After selecting the user, the system offers to enter a word for search (Fig. 6).





**Figure 7.** Start of the system

After entering a word, the system provides definitions in Ukrainian and English, as well as short descriptions and definitions. The system automatically offers to enter the next word for search and the user can perform the next search: Пошук значення слів (Search the word meaning) [Poshuk znachennia sliv] (Fig. 7)



**Figure 8.** Execution of the next stage

To go to the next section, the user can select the View list of categories section. After that, a choice of sections is offered according to the available parts of speech, namely Verbs, Nouns, Adjectives. Also, the user can go back one step using the appropriate button. After selecting the appropriate category, the user receives a list of possible words. After choosing a certain word, the user receives information in Ukrainian, English, as well as short descriptions and definitions (Fig. 8).

When the Back button is pressed, the user falls into the category selection menu: Переглянути список категорій (View list category) [Perehlyanuty spysok katehorii)]. Where you can perform a similar search by category Noun and Verb. When the Back button is pressed, the user drops into the Main menu to select the required operating mode. (Fig. 9).

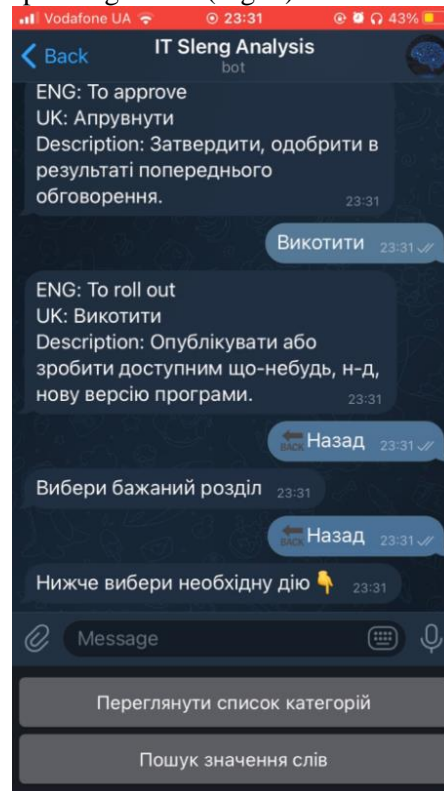


Figure 9. Return to the main menu

## 5. Conclusions

The findings of the study consider a development of the analysis system that enables users to obtain the meaning and the translation so to understand an IT slang word. The proposed system is aimed at word analysis of the IT field. The analysis shows that its primary features are search for the entered word definition or search for the word by selecting an appropriate category. We have shown that the systems function as a single-user application only and can be run on any device with an installed application/web version of the Telegram messenger. A chatbot's adoption has been hailed as a breakthrough in how people communicate with businesses and technology. NodeJS and Mongo DB were used in its development. Doc and Vocab are the central data in SpaCy. Doc was used to store sequences of tokens which are common sequences of characters found in text. Their annotations are also placed in Doc while Vocab stores reference tables. The model recognizes statistical relationship between tokens and excel at building the next tokens as a sequence. The program stands out in convenient interface and a search engine allows users to find word definition. The database keeps track of the user's progress towards completion and advises on the next steps.

Future research should consider adjoining additional feature as voice input kit to adopt audio recognition.

## 6. References

- [1] E. V. Stryha, Computer slang in Modern English: origion and function, in: Notes on Roman and Germanic philology, V 0, N. 1 (32), 2015, pp. 186-191. doi: 10.18524/2307-4604.2014.1(32).41545.
- [2] M. Dibrova, English for IT: words, slang and utterances which are helpful for your work, 2021. URL: <https://mc.today/blogs/anglijska-dlya-it-spetsialistiv-leksika-ta-frazi-shho-vam-stanut-u-nagodi/>.
- [3] Anglicism – IT slang, New IT School. URL: <http://www.itschool.vn.ua/anglicisms-it-slang/>.
- [4] M. Bogachuk, D. Bihunov, The structural-semantic features of computer terms in English, in: Cognitive Studies, V. 20, 2020. doi: 10.11649/cs.2262.
- [5] N. Polishchuk, Computer terminological vocabulary in mass media. In: Scientific Bulletin of Lesya Ukrainka, Eastern European National University, Philological sciences, Linguistics., N 1, 2013, pp. 95-100. URL: [http://nbuv.gov.ua/UJRN/Nvnuflm\\_2013\\_1\\_23](http://nbuv.gov.ua/UJRN/Nvnuflm_2013_1_23)
- [6] Y. Lynnyk, I. Navrotska, The outline of contemporary English and Ukrainian computer slang lexicon, in: Research trands in modern lingistics and literature, V. 5, 2022. doi: 10.29038/2617-6696.22.5.4.15.
- [7] T. Nikolaieva, Y. Boyko, O. Yemets, Y. Herasymenko, I. Dumchak, Slang vocabulary of the Ukrainian and English language: translation aspect, in: World Journal of English, V. 13, N. 5, 2023. doi: 10.5430/wjel.v13n5p52.
- [8] H. Shayner, IT slang in the modern Ukrainian language, in: Scientific overview, n 3 (75), 2021.
- [9] L. Stavvytska, Ukrainian jargon: Dictionary, 2005. URR: [https://chtyvo.org.ua/authors/Stavytska\\_Lesia/Ukrainskyy\\_zharhon\\_Slovnyk/](https://chtyvo.org.ua/authors/Stavytska_Lesia/Ukrainskyy_zharhon_Slovnyk/).
- [10] D. Gannon, Building a ‘ChatBot’ for scientific research, 2018. doi: 10.13140/RG.2.2.17641.39528.
- [11] Yu. A. Danko, Social networks as a form of modern communication: pluses and minuses, in: Modern society, 2012, Issue 2, pp. 179-184. URL: [http://nbuv.gov.ua/UJRN/cuc\\_2012\\_2\\_25](http://nbuv.gov.ua/UJRN/cuc_2012_2_25).
- [12] E. Adamopoulou, L. Moussiades, An overview of Chatbot Technology in Artificial Intelligence, in: Application and Innovations, AIAI, 2020, IFIP Advances in Information and Communication Technology, V. 584, 2020. doi: 10.1007/978-3030-49186-4-31.
- [13] S. Anzaroot, A. McCallum, UMass citation field extraction dataset, 2013. URL: <http://www.iesl.cs.umass.edu/data/data-umasscitationfield>.
- [14] Ch. Misischia, F. Poecze, Ch. Strauss, Chatbots in customer service: Their relevance and impact on service quality. in: Procedia Computer Science, V. 201, 2022. Doi: 10.106/j.procs.2022.03.055
- [15] G. Caldarini, S. Jaf, K. A. McGarry, Literature Survey of recent advances in chatbots, N: Information, 13 (1), 2022. doi: 10.3390/info13010041.
- [16] T. Kudo, M. Ishino, K. Saotome, W. A. Kataoka, Proposal of transaction processing methods for MongoDB, in: Procedia Computer Science, Volume 96, 2016. doi: 10.1016/j.procs.2016.08.251
- [17] D. Chauhan, K. Bansal, Using the advantages of NOSQL: a case study on MongoDB, in: International journal on recent and innovation trends in computing and communication, V. 5, N. 2, 2017. doi: 10.17762/ijritcc.v5i2175
- [18] Use case diagram. URL: [https://desosa.nl/projects/spacy/2020/04/09/images/spacy/spacy\\_architecture.png](https://desosa.nl/projects/spacy/2020/04/09/images/spacy/spacy_architecture.png)
- [19] W. Hasyim, S. Pramono, Web-based Telegram Chatbot Management System: Create Chatbot without programming language requirements, IOP Conference Series: Materials Science and Engineering, 2021. doi: 10.1088/1757-899X/1096/1/0120075
- [20] E. Adamopoulou, L. Moussiades, Chatbots: History, technology and applications, in: Machine Learning with Applications, V. 2, 2020. doi: 10.1016/j.mlwa.2020.100006