UDC 004.58+004.94
DOI: 10.15587/1729-4061.2024.304048

CONSTRUCTION OF A MODEL FOR MATCHING USER'S LINGUISTIC STRUCTURES TO A CHAT-BOT LANGUAGE MODEL

Olha Kryazhych
Corresponding author
PhD, Senior Researcher, Associate Professor*
Institute of Telecommunications and Global Information Space of the National Academy of Sciences of Ukraine
Chokolivskiy blvd., 13, Kyiv, Ukraine, 03186
E-mail: economconsult@gmail.com

Oleksandr Vasenko
PhD, Associate Professor, Head of Department*

Liudmyla Isak
Senior Lecturer*

Ihor Havrylov
PhD student*

Yevhen Gren
PhD student*

*Department of Digital Learning Technologies
Hryhorii Skovoroda University in Pereiaslav
Sukhomlynsky str., 30, Pereiaslav, Ukraine, 08401

The research object of this work is the linguistic structures of the user when constructing a request to a chatbot with generative artificial intelligence. The study solved the task of improving the communication mediation algorithms of chatbots through the comparison models of linguistic structures by users. Sometimes the user intentionally or due to lack of information forms an inaccurate request. Formally, this is described by logical operations "And" and "And or Not".

As a result of the research, a model was built comparing linguistic structures at the input with the information model of the response at the output. The model was based on an approach with recursive creation of an answer. That has made it possible to determine the basic characteristics of the object of the request and form an answer on this basis. Using this approach improved the accuracy of the chatbot response. It also made it possible to consider the linguistic structure of the user through its formalization. The use of logic algebra made it possible to find typical errors of users during dialogs with generative artificial intelligence.

A feature of the reported advancement is that the comparison of models of linguistic structures of query formation is carried out through a recurrent algorithm. As a result, it makes it possible to compare the query in such a way as to reduce the absolute error of the primary data by 0.02 % and simplify the process of mathematical calculations. At the same time, the received information becomes more accurate – the number of references increases from 2 to 6 sources.

The proposal could be used in practical activities to improve the natural language recognition technologies of users in chatbots with generative artificial intelligence. On this basis, it is possible to devise various applications and services for training and practical activities

Keywords: generative artificial intelligence, recurrent algorithm, formalization of user request, basic sequence

Received date 04.03.2024
Accepted date 13.05.2024
Published date 28.06.2024

***How to Cite**: Kryazhych, O., Vasenko, O., Isak, L., Havrylov, I., Gren, Y. (2024). Construction of a model for matching user's linguistic structures to a chat-bot language model. Eastern-European Journal of Enterprise Technologies, 3 (2 (129)), 34–41. https://doi.org/10.15587/1729-4061.2024.304048*

1. Introduction

The number of users of chatbots with artificial intelligence (AI) is growing along with the expansion of the scope of application of these specialized computer programs for human interaction. The number of users of AI applications is growing exponentially [1]. Owing to the use of machine learning approaches in chatbots, it is necessary to constantly form arrays of data regarding requests, for the purpose of further improvement and the formation of adequate answers to user questions.

However, the use of incorrect logical structures, the construction of a query based on abstraction, the formulation of statements in a conversational style, distance the user from receiving an accurate and adequate answer. In this case, the chatbot's lack of reasoned traffic of knowledge leads to an information limitation in the process of forming an answer and the inability of generative algorithms to provide answers to more complex questions [2].

The latter has a significant impact when using chatbots in professional training or business. Therefore, study [1] distinguishes two types of use of chatbots – dominance and understanding. And the users of chatbots with AI algorithms are divided into those who quickly give up work when requests fail, pragmatists, progressive, and persistent. Each user, regardless of type, can formulate a request according to his/her own linguistic structure. The chatbot must recognize the request, identify the object of the request, generate a response, and issue it to the user. To this end, it is necessary to implement certain models, according to which, when recognizing the request object, data arrays are accessed. These arrays, due to self-similarity [3] with the help of previously defined features, make it possible to recursively reproduce not only the features of the object but also the entire class. That is, to form a special structure for grouping related variables and functions, taking into account their properties [4]. Similar recursive models [5] of creating a response to a user's request can become the basis for machine learning of chatbots to adequately respond to any request. In addition, with a recursive model, the chatbot will "understand" requests in other natural languages, using slang or a mixture of languages. In this case, an infinitely large number of objects will be recursively reproduced according to the query keyword, using AI access to existing knowledge bases.

The above proves the relevance of the research topic, which requires study and analysis to improve the communication mediation algorithms of chatbots. The development of models for comparison of users' linguistic structures will make it possible to increase the quality of knowledge in the knowledge bases of chatbots. It can also contribute to improving the modification of knowledge bases in the process of work and quick adaptation to the problem area.

2. Literature review and problem statement

Work [6] touches on deep learning of AI and proposes a computational process for solving it, which is a rather traditional solution for this class of problems. The advantages of research are the possibility of iterations and coding at each step. They are implemented through a universal transformer. It acts as a structure for the simultaneous integration of the two mentioned processes owing to the continuous transformation of self-awareness in the depth of the position. The disadvantage is the complication of the implementation of calculations for building the model. For example, based on iteration (Radl+) in Haskell executables. However, the idea of using the Haskell language to solve the problems of training programs with the use of AI is also not new and is presented in detail in papers [7, 8].

Analysis of work [9] allows us to consider the evolution of AI chatbots through models starting from the Turing machine. And the main problem is precisely the accuracy of the answer based on the analysis of the user's request. The advantages of the presented developments in the work are detailed. The disadvantage is insufficient processing of language structures of the user. In work [10] it is noted that recognition of the user's emotions on request based on a consistent mediation model improves response results. This can be understood as an advantage. However, this requires improved chatbot communication models and reproduction of artificial emotions based on query analysis.

Paper [11] examines interaction with the user according to five models. These are Information System Success, Technology Acceptance Model, Kinship Theory, Coolness Theory, and Posthumanism. The positive result of the research is the statistically confirmed demand of users to receive relevant, reliable, short information in the shortest possible time. The problem of the research is the insufficiently developed mathematical apparatus for comparing requests and answers.

Work [12] analyzes the work of chatbots ChatGPT-3.5 and ChatGPT-4 based on a large language model (LLM). The advantages of the work are the details of LLM features. However, LLM is considered only through general purpose tasks with an extension to perform applied family medicine tasks. In study [13], the specified LLM model is proposed for the training of doctors. The advantage of the work is the definition of linguistic structures relative to the specialty. The disadvantage is that the variability of requests for communication with users in natural language is not considered.

An experiment with a discrete selection of characteristics to create a chatbot learning model is reported in [10]. The advantage of the work is the combination of the LLM model with operational design models. The result of the work was the development of a chatbot for training based on generative AI. The disadvantage is that the rules of possible linguistic structures were not defined in advance.

Work [14] proves the inability of large language models to provide meaningful answers to the user's request. Such a definition is an advantage of the work. And although the answers are not imprecise, the algorithms of the LLM model bypass questions for which certain restrictions are set. That is, the lack of models for determining the linguistic structures of users is a drawback.

The advantages of study [15] are the indication of existing problems of interaction of chatbots in the fact that during a digital conversation, the user's requirements are not precisely defined. Therefore, the answers may not be relevant to the request. The authors propose to match requests and responses using a bidirectional recurrent neural network with a fuzzy naive Bayes classifier (BRNN-FNB). The combination of machine learning and natural language processing allows for accurate results. The disadvantage is not taking into account the need to increase the accuracy of the answer. And this is precisely what the improvement of the process of calculating the model of the linguistic structure of the query needs.

This allows us to state that it is appropriate to conduct a study aimed at building a mathematical model of the comparison of the construction of the request with the model of the formation of the answer. In such a model, the basis should be the turning of the posed question to oneself. That is, when defining the request object, its basic simple characteristics and application cases are defined. On this basis, a function is built that describes the logical structure of the request. Such a formalization will make it possible to refer to the knowledge bases operated by the AI chatbot to form arrays of information from which the answer will be generated.

3. The aim and objectives of the study

The purpose of our study is to build a model of comparing linguistic structures at the input with the information model of the response at the output of the chatbot. This will make it possible to recursively create responses to the user's request, with the definition of base cases regarding the request object and the analysis of the logical structure of the request.

To achieve the goal, the following tasks were set:

– to develop an algorithm for building a user request to a chatbot based on defined linguistic structures in natural language;

– to build a linguistic model of the user request and to compare it with the language model of the chatbot based on recurrent relations.

4. The study materials and methods

The research object of our work is the linguistic structures of the user when constructing a request to a chatbot with generative artificial intelligence. A language structure is understood as phrases in the user's natural language, combined by grammatical and stylistic links, used to create a request to the chatbot.

The main hypothesis of the study is the following: as the language structure of the request approaches the language model of the chatbot, the accuracy of the response to the user increases. Users construct queries based on their own reasoning, emotions, knowledge of natural language and level of education. Accordingly, in the process of working in a “man-machine” tandem, the task of matching language structures must be solved.

The work assumes that the user intentionally or due to lack of information forms an inaccurate request to the chatbot. Formally, this is described by logical operations when forming “And” and “And or Not” queries. Simplification is a generalized consideration of the language model without specification to the request of a specific topic. It is also a simplification to use basic trigonometric functions to describe the linguistic structure of the user.

The limitation of the work is the use of one chatbot for research – GitHub Copilot with a generative AI that uses a modified version of GPT-3. In particular, Bing Copilot and extensions for the Google Chrome browser were used. Some aspects of the research were tested in the work with ChatGPT and Gemini. However, meaningful studies of the possibilities of using the user’s linguistic structures with chatbots ChatGPT and Gemini have not been conducted.

A user’s linguistic construct may contain standard keywords accompanied by definitions, terms, and action descriptions. The user uses natural language expressions in communication with the chatbot. This can lead to an approximate definition of recognized request objects when formalizing the request through recursion.

Recursion, as a method of defining a class or object, is considered through the preliminary assignment of basic cases, based on which the rule for constructing this class or object is set [5]. Recursion is a process for a procedure, where the specification of its execution takes place at its own step for the implementation of another procedure. Considering the execution procedure as an algorithm according to defined rules, some self-similarity can be observed. It consists in breaking down the given task into given steps. In this case, recursion can be considered as a pattern, which is described by isolated fragments – separated subtasks of one task.

By implementing a similar task with the help of functional programming, one can get a reliable model implementation process with recursive reproduction of the object. This is due to the mathematical transparency of the functional programming paradigm. However, as noted in [5], directly creating a functional program related to recursive reproduction based on specifications is a difficult task. Most existing conversion methods refine specifications into programs at the abstract algorithm level. That is, most implementations of recursions are based on cycle invariants.

A formalized user request can be represented by some sequence, without a loop. The sequence will mean the number of a natural language word in the domain thesaurus. In this case, one can use the method of calculating values based on previous members of the sequence to calculate infinite products [16]. This method makes it possible to perform calculations with variable precision without using a reduction to a given interval, which reduces the time of information processing. This is due to the fact that the use of recurrent relations makes it possible to shorten the initial interval several times, and, accordingly, to reduce the number of iterations.

Using the approximation with a multiple decrease of the interval, one can use initial or final approximations. And the effectiveness of building a user language model should be considered through the error criterion. In general, such a model can be written as:

$$\Delta \approx C_n\left(\frac{x}{N^m}\right)^n \approx \frac{x}{N^{mn+l}},$$

where C_n are constants that mostly depend on the parameter n;

N is a number that determines the amount of interval reduction;

m is the number of iterations using the recurrent formula;

n is the order of the term of the expression discarded during approximation during the initial or final approximation of the recurrent relation $l=-\log_N|C_n|$.

Based on this model, it is possible to determine the values of parameters m and n. And the reduction of the interval during approximation will be proportional to the value of $1/N^m$. An increase in the value of N leads to the complication of recurrent formulas, in connection with which the definition of approximation in this work can be used with the prefix R_m. In addition, as N increases, the possibility of parallelizing recurrent relations based on R_m-approximation increases. To investigate this, the initial error model can be represented as:

$$\Delta \approx C'_n\left(\frac{x}{\left(N+\Delta N\right)^{m+\Delta m}}\right)^{n+\Delta n},$$

where ΔN, Δm, Δn are increments of the corresponding parameters N, m, and n.

The logarithm of this expression:

$$\ln\Delta \approx \ln C'_n + (n+\Delta n)\left(\ln x - (m+\Delta m)\ln(N+\Delta N)\right).$$

It follows from the latter that an increase in the values of m and n affects the reduction of the error . But with serial and parallel calculations, the algorithms for selecting parameters will differ. Therefore, it should be noted that an increase in the parameter m leads to an increase in the number of iterations according to the recurrent formula and an increase in error. An increase in the parameter n will lead to an increase in the number of terms in the initial or final approximation, as well as a decrease in C'_n. Therefore, for each fixed N, it is necessary to choose the ratio between m and n.

For recurrent relations, the concept of basic sequence should be introduced. The basic sequence of recurrent relations is a sequence of recurrent formulas whose p-term has a parameter n_p greater than n_{p-1} in the expression $R(nx_m)=f[R(x_m)]$, where $x_m=x/n^m$.

We give examples of such sequences.

To calculate the sin$x$ function:

$$R_{m-1}=\sum_{k=0}^{n}(-1)^k C_{2n+1}^{2k+1} Z_m^{2k+1}\left(1-R_m\right)^{n-2k},$$

where $R_m=\sin\left[x/(2n+1)^m\right]$, $m=m_0, m_0-1, ..., 0$, hence $R_0=\sin x$.

This can be represented in a general way:

$$R_{m-1}=T_{2n+1}\left(R_m\right),$$

where $T_{2n-1}(R_m)$ is the Chebyshev polynomial.

If the recurrence relation $R_m=2R_{m-1}$ $m=\overline{n,1}$, is used for the function $y=e^x$, then the absolute error caused by the rounding error will be equal to:

$$\Delta_n = 2^n\Delta_0\left(e^{x/2^n}+\overline{\Delta_0}\right)^{2n-1},$$

where $\left|\overline{\Delta_0}\right| \le \Delta_0$, and the x value is determined from the equation $e^x=2^p l^x$.

That is, the error can be greater than 2^{n-t-1}, where t is the bit size of numbers. If instead of the function $y=e^x$ we consider the function $v=e^x-1$, then the error value will not be greater than the value $O(n2^{-t})$. Such features were used in the construction of the model.

The most used words in search engines were the input data for the formation of arrays of words on a separate topic. To this end, words were chosen on the topics of programming, software engineering, and robotics. Arrays were supplemented with data from ontologies on the specified topic.

Statistical information was used to build the models. The response time to the request from the chatbot, the accuracy of the answer were recorded. On this basis, arrays of words were formed, with the use of which, when answering a request, the greatest error was observed. Processing of linguistic structures was carried out on the basis of content analysis. Statistical methods of information processing were used to calculate the uniform and exponential distribution of the accuracy of the answer when using the keyword. Given that relatively small data sets were used in the work, processing took place in Excel.

Hardware for our work is personal computers with an AMD Ryzen 7 5800X processor. RAM DDR4-3200, 8–16 GB. The operating systems are Windows 10, 11.

Software for conducting computer experiments and research results were developed in the form of applications. A Haskell application was created to select words from an array to construct a user query. A C++ application was designed for model calculation and visualization. Bing Copilot chatbot software was used for our work. It was applied to test users' linguistic structures to collect statistical data on response accuracy.

5. Results of research on the development of a comparison model

5. 1. Algorithm for constructing a user request to a chatbot

To develop a recurrent algorithm using the R_m-approximation, the mathematical representation of the R_m-function should be introduced. This makes it possible to bring a query created in natural language closer to the formalized criteria perceived by the chatbot.

The R_m-function is understood as a direct or inverse recurrent relation of the form:

$$R_{m+1} = f(R_m), \tag{1}$$

or:

$$R_{m-1} = f(R_m). \tag{2}$$

The first step of the algorithm is some initial approximation R_0, and the function R_m is the desired function, as given in (1). In case (2), some initial approximation R_{m_0}, is given, and the function R_0 is the desired function. Formula (2) can be derived from an expression of the form:

$$(x_m / n) = f\left[R(x_m)\right], \tag{3}$$

by substituting:

$$x_m = \frac{x}{n^m}, \tag{4}$$

when determining R_m as:

$$R_m = R(x_m), \tag{5}$$

where n can be represented as the base of the calculation system.

Formula (5) can be derived from the expression:

$$R(nx_m) = f\left[R(x_m)\right], \tag{6}$$

by replacing (4) with the adopted definition (5).

As an initial approximation for (2), we can take several terms of the expansion of R_m function in the Taylor series.

To implement the second step in order to estimate the absolute error of the method, formulas (1) or (2) should be supplemented with:

$$R_m = \overline{R_m} + \overline{\Delta_m}, \tag{7}$$

where $\overline{R_m}$ is the exact value of R_m;

Δ_m – absolute error of R_m.

The third step makes it possible to get an estimate of absolute error:

$$\overline{\Delta}_{m+1} \le k_1 \overline{\Delta}_m, \tag{8}$$

or:

$$\overline{\Delta}_{m-1} \le k_2 \overline{\Delta}_m, \tag{9}$$

where k_1, k_2 are constants.

It follows from expression (8) that the estimate of the absolute error of the method in the case of a direct recurrent sequence takes the form:

$$\overline{\Delta_m} \le k_1^m \overline{\Delta}_0, \tag{10}$$

From expression (9), it can be concluded that the estimate of the absolute error of the method in the case of an inverse recurrent sequence takes the form:

$$\overline{\Delta_0} \le k_2^m \overline{\Delta}_m. \tag{11}$$

At the fourth step of the algorithm, the errors of the initial data are evaluated. This can be represented as:

$$\overline{\Delta}_0 \le p,$$

where p is a constant.

As a result, from (10) we can get:

$$\overline{\Delta_m} \le k_1^m p. \tag{12}$$

For the inverse recurrence sequence of (11):

$$\overline{\Delta}_0 \le k_2^m p. \tag{13}$$

The resulting model can be represented in the form:

$$R(\phi(x_m)) = f\left[R(x_m)\right],$$

where $x_m=\varphi^{-1}(x)$, $\varphi^{-1}(x)$ is the function mutually inverse to the function $\varphi(x)$.

5. 2. Building a linguistic model of the user's request to the chatbot

A user query that can be described in terms of algebraic logic operations on some nonempty set – negation, conjunction, and disjunction. In this case, the query can be formalized through the function y=cos x. A query can be described in terms of the truth-seeking operation "And" and "And or Not". In this case, the correct answer to such a request can be obtained by the given recurrent algorithm through the approximation of cos x based on the harmonic polynomial $H_n^{(0)}$:

$$H_n^{(0)}\left(\cos x_m, \cos x_m\right) = \cos nx_m = \\ = \sum_{k=0}^{\left[\frac{n}{2}\right]} (-1)^k C_n^{2k} \left(1-\cos^2 x_m\right)^k. \tag{14}$$

In (14), one can make a substitution according to the first step of the given algorithm using (4) and taking into account (5). As a result, a recurrent model for calculating cos x will be derived:

$$R_{m-1} = \sum_{k=0}^{[n/2]} (-1)^k C_n^{2t} R_m^{n-2k} \left(1-R_m^2\right)^k, \tag{15}$$

hence, $\cos x = R_0$.

But individual cases (15) are possible when n=2, 3, 4, 5.

When n=2, the recurrent formula will look like this:

$$R_{m-1} = 2R_m^2 - 1, \tag{16}$$

where $R_m=\cos(x/2^m)$, and $\cos x=R_0$.

If one implements the second step of the algorithm, and enters (7) to (16), one can get:

$$\bar{R}_{m-1} + \bar{\Delta}_{m-1} = 2\left(\bar{R}_m + \bar{\Delta}_m\right)^2 - 1 = \\ = 2\bar{R}_m - 1 + 4\bar{R}_m \bar{\Delta}_m + \bar{\Delta}_m^2 = \bar{R}_{m-1} + 4\bar{R}_m \bar{\Delta}_m + \bar{\Delta}_m^2.$$

And taking into account that for a sufficiently large m, the quantity $\bar{R}_m \approx 1$, and for all $\bar{m}$ the quantity $\bar{R}_m \leq 1$, and also neglecting the term $O\left(\Delta_m^2\right)$, one can represent as:

$$\left|\bar{\Delta}_{m-1}\right| \leq 2^2 \bar{\Delta}_m. \tag{17}$$

From here, the estimation of the absolute error of the method will look like this:

$$\left|\bar{\Delta}_{m-1}\right| \leq 2^{2m} \bar{\Delta}_m. \tag{18}$$

For an initial approximation, one can take two terms of the expansion of the R_{m0} function in the Taylor series:

$$R_{m_0} = 1 - x^2 / 2^{2m+1}. \tag{19}$$

Provided that $x \in \left[0, \frac{\pi}{2}\right]$ the estimation of the value of the absolute error from the initial data with calculation (19) will take the form:

$$\Delta_{m_0} = \frac{x}{3!2^{3m}} < \frac{1}{2^{4m+1}}. \tag{20}$$

And taking into account inequalities (18), (20), we can write:

$$\Delta_0 < 1/2^{2m+1}. \tag{21}$$

We can use as an initial approximation the three terms of the expansion of the R_m function in the Taylor series:

$$R_{m_0} = 1 - \frac{x^2}{2^{2m+1}} + \frac{x^4}{3} \cdot 2^{4m+3}. \tag{22}$$

Then for $x \subset [0, \pi/2]$ the following holds:

$$\left|\overline{\Delta_{m0}}\right| \approx \frac{x^6}{6!} 2^{6m} < \frac{1}{2}^{6m+4}, \tag{23}$$

hence:

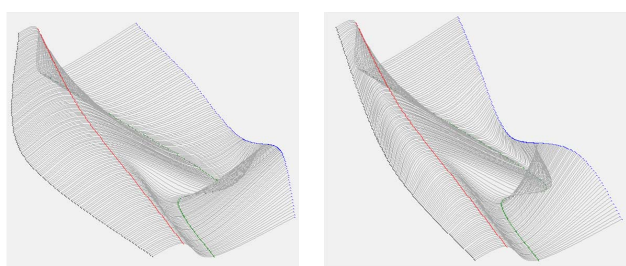
$$\left|\bar{\Delta}_0\right| < \frac{1}{2^{4m+4}}. \tag{24}$$

It can be seen from model (24) that the number of iterations m decreases compared to (19). From the recurrent formula (15) for calculating $\cos x$ at n=2, 3, 4, 5, it can be seen that the value of the constant K_2 in formula (9) takes the form $K_2=n^2$. Given this, formulas (11), (13) can be written as follows:

$$\left|\bar{\Delta}_0\right| < n^{2m} \bar{\Delta}_m, \; \left|\bar{\Delta}_0\right| < n^{2m} p.$$

These formulas are a representation of the estimate of the absolute error of the method.

An application in the Haskell language was used to implement the computer experiment. Samples of 3–6 words were created from an array of 2,000 words formed according to a separate topic. Users (100 people) with selected words created queries to chatbots in natural language using logical operations "And" and "And or Not". 1.2 thousand requests were created. For the construction of linguistic models, 0.2 thousand queries were used for each specified logical operation. To build the models, the queries leading to the answer with the greatest degree of error were selected. Visualization of the linguistic model at the request of the user using the recurrent algorithm is performed in the C++ application and shown in Fig. 1.



a *b*

Fig. 1. Visualization of linguistic models based on the user's request, performed in natural language using logical operations: *a* – user's language model when creating a request using the logical operation "And"; *b* – user's language model when creating a request using the logical operation "And or No"

Calculation of the model through the harmonic polynomial $H_n^{(0)}$ (14) makes it possible to approximate the query based on the linguistic model of the user to the formalized language model understood by the chatbot algorithms. The results of the visualized model of the user's language according to the specified logical operations are shown in Fig. 2.

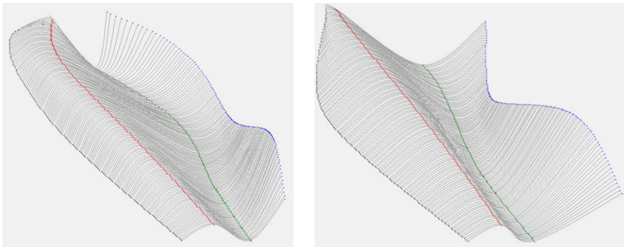


Fig. 2. Visualization of linguistic models after R_m-approximation when creating a request to a chatbot: *a* — a model of the user's language when creating a request using the logical operation "And"; *b* — user's language model when creating a request using the logical operation "And or No"

As a result of the calculation of the model through the harmonic polynomial, the value of the absolute error of the primary data is reduced by 0.02 %. Three terms of the expansion of the R_m function into a Taylor series were used as an initial approximation. The number of calculation iterations did not exceed 4. In addition, an interesting result was obtained when testing queries using improved linguistic structures of users. The information in the response became more accurate, and the number of references to sources from the chatbot increased from two to six on average.

6. Discussion of results of the matching model study

During the development of the algorithm for building the user's request to the chatbot, a special evaluation of the errors of the initial data was carried out. It is these mistakes when creating a query in the user's natural language that can lead to an inaccurate answer.

Visualization of linguistic models by user request (Fig. 1, *a*, *b*) is a model of displaying a request made by a user trying to find an answer. At the same time, the user uses natural language. In the experiment, the condition was set that the user does not understand the essence of the question and tries to construct a query using repetitions and objections, which can be described by logical operations. Each horizontal curve describes a request. The vertical curves hold the query keyword points. This is an indication that the request was made within the given limits. The drawings are obtained using a developed application using a 3D visualization tool.

Using the algorithm reported in our work, models of linguistic structures of requests were built. Analyzing the built models by logical operations, it can be noted that the least accurate answers were obtained when creating cycles in requests. In Fig. 1, *a*, *b*, it is represented in the form of "pockets". For example, repeating the logical operation "And" (Fig. 1, *a*) led to an unclear answer. On the model, this can be seen in the formation of identity – the superimposition of concepts one on top of the other. In response, the chatbot provided a list of the listed with definitions. The logical operation "And or No" (Fig. 1, *b*) with several repetitions in one request led to the inconsistency of the answer and the formation of repeated retelling when forming the answer. As a result, the user received irrelevant facts or information that had little to do with the subject of the request. The results regarding the occurrence of "pockets" are explained by the peculiarities of the applied logical structures, when the user does not highlight the object of the request, does not emphasize its meaning.

The results of the approximation by the logical operation "And or No" made it possible not to avoid but to explain questions with certain restrictions in response to the request, supplementing work [15]. This is due to the use of recursion, based on the example of [5]. At the same time, the difficulty of the calculations, in contrast to work [6], is reduced due to the possibility of setting the number of iterations of the calculation.

The constructed linguistic model of the user request and its comparison with the language model of the chatbot made it possible to find a certain parity. That is, in the future it will make it possible to develop rules of communication with chatbots for users to receive adequate answers. It also makes it possible to improve algorithms for "understanding" requests by chatbots on the basis of recurrent relationships. This follows from our results. Thus, analyzing Fig. 2, it can be noted that using the initial approximation (19), it is necessary to select the number of iterations m from requirements (21). That makes it possible to increase the accuracy of calculations. In addition, to calculate $\cos x$ in formulas (16), (19), it is necessary to keep an additional 2_m binary digits, or the corresponding number of decimal digits. In the case when the introduction of a large number of additional digits is undesirable, formula (16) can be converted by introducing an additional normalizing factor 2^{2m}.

Fig. 2 shows an approximation of the user's linguistic structure to the linguistic model of the chatbot. This was done by analyzing the total array of the thesaurus by topic and words that led to incorrect answers.

The result shown in Fig. 2, *a* allows us to supplement the result of work [15] by building an approximate model with clarification of the request. The logical operation "AND" in this case is harmonized to a logical unit. That is, the user request model limits non-essential repetitions when searching for the truth. These repetitions can be used in clarifying queries by topic to develop communication models of chatbots [10] owing to the application of a recurrent algorithm.

Our solutions regarding the construction of the combination model solve the task identified during the review of primary sources. This is achieved by combining logical operations with a recursive algorithm. As a result, it makes it possible to "approach" the query in natural language to LLM models, improving work [14] owing to the results shown in Fig. 2.

The limitations of this work are the use of two logical operations. This does not make it possible to analyze the linguistic model of the user when the logical structure of statements changes. In addition, the limitations include specifying the number of calculation iterations. This simplifies the calculation process and its algorithmization. However, this can lead to approximation errors when using large information arrays.

The disadvantage of the work is the small volume of the array for selecting words for the purpose of creating a request to the chatbot. Also, the number of conducted tests and the sample size for creating models can be considered a disadvantage. That is why the prospects for our studies may involve the construction of models based on various logical operations. At the same time, linguistic structures that are described by various functions should be used. Including non-linear ones.

The developed models and algorithm can be used to improve chatbot technology with generative artificial intelligence for machine learning. Also, the results can be used to develop applications that use services with artificial intelligence. This can be used by users for training, finding practical solutions, and expanding the functions of specialists in remote work.

7. Conclusions

1. A recurrent algorithm for constructing a query to a chatbot based on defined linguistic structures in the user's natural language has been developed. A feature of this algorithm is the gradual approximation of the user's request to the best understanding of it by the chatbot algorithms. The peculiarities of this algorithm were studied by the number of iterations with the evaluation of the errors of the initial data. In the end, this allows us to assess the absolute fallibility of the method for constructing a query to the chatbot.

2. A linguistic model of the user's request was built and compared with the language model of the chatbot. The model is built on the basis of the developed recurrent algorithm. The combination of logical operations with a recurrent algorithm makes it possible to approach user requests with clarification of the result-answer. The conducted comparison of the linguistic structure of the user and the language model of the chatbot made it possible to achieve a decrease in the value of the absolute error of the primary data by 0.02 %. The number of iterations during calculations did not exceed 4.

Conflicts of interest

The authors declare that they have no conflicts of interest in relation to the current study, including financial, personal, authorship, or any other, that could affect the study and the results reported in this paper.

Funding

The study was conducted without financial support.

Data availability

All data are available in the main text of the manuscript.

Use of artificial intelligence

The authors used artificial intelligence technologies within acceptable limits to provide their own verified data, which is described in the research methodology section.

References

1. Gkinko, L., Elbanna, A. (2023). The appropriation of conversational AI in the workplace: A taxonomy of AI chatbot users. International Journal of Information Management, 69, 102568. https://doi.org/10.1016/j.ijinfomgt.2022.102568
2. Hauptmann, C., Krenzer, A., Völkel, J., Puppe, F. (2024). Argumentation effect of a chatbot for ethical discussions about autonomous AI scenarios. Knowledge and Information Systems. https://doi.org/10.1007/s10115-024-02074-x
3. Auffarth, B. (2020). Artificial intelligence with python cookbook: proven recipes for applying ai algorithms and deep learning techniques using tensorflow 2.x and pytorch 1.6. PACKT Publishing, 468.
4. Working Draft, Standard for Programming Language C++. Available at: https://www.open-std.org/jtc1/sc22/wg21/docs/papers/2017/n4713.pdf
5. Zuo, Z., Zeng, Z., Su, W., Huang, Q., Ke, Y., Liu, Z. et al. (2023). Specification transformation method for functional program generation based on partition-recursion refinement rule. Information Sciences, 633, 613–632. https://doi.org/10.1016/j.ins.2023.03.055
6. Bian, W., Li, C., Hou, H., Liu, X. (2023). Iterative convolutional enhancing self-attention Hawkes process with time relative position encoding. International Journal of Machine Learning and Cybernetics, 14 (7), 2529–2544. https://doi.org/10.1007/s13042-023-01780-2
7. Sannella, D., Fourman, M., Peng, H., Wadler, P. (2021). Introduction to Computation. In Undergraduate Topics in Computer Science. Springer International Publishing. https://doi.org/10.1007/978-3-030-76908-6
8. Bossard, A. (2022). The SOF Programming Paradigm. International Journal of Software Innovation, 10 (1), 1–14. https://doi.org/10.4018/ijsi.309965
9. Nirala, K. K., Singh, N. K., Purani, V. S. (2022). A survey on providing customer and public administration based services using AI: chatbot. Multimedia Tools and Applications, 81 (16), 22215–22246. https://doi.org/10.1007/s11042-021-11458-y
10. Park, G., Chung, J., Lee, S. (2022). Effect of AI chatbot emotional disclosure on user satisfaction and reuse intention for mental health counseling: a serial mediation model. Current Psychology, 42 (32), 28663–28673. https://doi.org/10.1007/s12144-022-03932-z
11. Niu, B., Mvondo, G. F. N. (2024). I Am ChatGPT, the ultimate AI Chatbot! Investigating the determinants of users' loyalty and ethical usage concerns of ChatGPT. Journal of Retailing and Consumer Services, 76, 103562. https://doi.org/10.1016/j.jretconser.2023.103562

12. Huang, R. S., Lu, K. J. Q., Meaney, C., Kemppainen, J., Punnett, A., Leung, F.-H. (2023). Assessment of Resident and AI Chatbot Performance on the University of Toronto Family Medicine Residency Progress Test: Comparative Study. JMIR Medical Education, 9, e50514. https://doi.org/10.2196/50514
13. Xie, Y., Seth, I., Hunter-Smith, D. J., Rozen, W. M., Seifman, M. A. (2023). Investigating the impact of innovative AI chatbot on post-pandemic medical education and clinical assistance: a comprehensive analysis. ANZ Journal of Surgery, 94 (1-2), 68–77. https://doi.org/10.1111/ans.18666
14. Kuckelman, I. J., Yi, P. H., Bui, M., Onuh, I., Anderson, J. A., Ross, A. B. (2024). Assessing AI-Powered Patient Education: A Case Study in Radiology. Academic Radiology, 31 (1), 338–342. https://doi.org/10.1016/j.acra.2023.08.020
15. Dongbo, M., Miniaoui, S., Fen, L., Althubiti, S. A., Alsenani, T. R. (2023). Intelligent chatbot interaction system capable for sentimental analysis using hybrid machine learning algorithms. Information Processing & Management, 60 (5), 103440. https://doi.org/10.1016/j.ipm.2023.103440
16. Noroozi, N., Mironchenko, A., Wirth, F. R. (2022). A relaxed small-gain theorem for discrete-time infinite networks. Automatica, 142, 110363. https://doi.org/10.1016/j.automatica.2022.110363