Virtual assistant constructing method for metrological measurements

D A Trokoz1,3, M A Mitrohin3, M P Sinev2, S A Zinkin3, V V Boriskin2, D O Neshko2 and N O Sivishkina2

1 Penza State Technological University, 1a, Baidukova Avenue / 11, Gagarina Street, Penza, 440039, Russian Federation
2 Penza State University, 40, Krasnaya Street, Penza, 440026, Russian Federation
3 E-mail: dmitriy.trokoz@gmail.com

Abstract. The article considers a dialog service which performs a bot assistant functions for recording metrological measurements results. The project conducted research of one of the modern approach to developing chatbot — rule-based chatbot. The main approach’s features were considered and compared with analogous bot assistant in that article. The aim of the research is development of the dialog service model to ease and accelerate metrological measurements. As a result, the article proposed recommendations for developing a dialog system which performs a bot assistant with functions of performing and recording metrological measurements results typed by the user to the chat or using a voice assistant.

1. Introduction
The subject to this research is a dialog service which serve as a bot assistant functions for performing metrological measurements. The bot assistant receives and processes user’s requests and controls dialogue based on a scenario.

The aim of the research is development of the dialog service model to ease and accelerate metrological measurements.

The study covered such organization of dialog system architecture as rule-based approach and described a possible chatbot messages structure for measurement’s optimization.

2. Rule-based chatbot
Over the past decade, programming of chatbots has received special advancement. Chatbots are capable of embrace a simple conversation, solving ordinary issues, as well as simplifying the performance of both everyday tasks and highly specialized work.

Scope of chatbots is extensive and chatbot’s a possible feature set depends on the field of application a lot. A common feature of each chatbot is that a chatbot is a virtual machine. It programmed to conduct a dialogue with one or more users. It means that chatbot is natural language understanding and natural language generating program. The most widespread dialog system generates an answer to the user’s question by comparison request with the existing questions in the database [1]. The algorithm called «rule-based» or «decision-tree». It consists in selecting and codification of important semantic parts of the sentence. Possible scenarios based on selecting parts are described in special formal script languages. Most common chatbot assistants implemented on rule-based approach [2].
Rule-based chatbots are implemented according to the next way:

- Developing of dialogue-trees and summary templates;
- Extracting user’s request parameters and filling them in variables by named entity recognition;
- Connection and passing parameters by chatbot to external services such as 1C, SAP, CRM, Deezer and so on; As a result, chatbot receives the date to generate a response [3, 4].

Modern rule-based development environments include such elements as:

- Rating scheme of parsing hypotheses;
- Named entity recognition in text;
- Morphological analysis and phrase analysis;
- Opportunity to integrate with external systems.

AIML (Artificial Intelligence Modelling Language) is an XML Artificial intelligence (AI) markup language for creating natural language software agents or rule-based chatbots. It contains processing rules for source date. The main AIML’s drawback is that methods of extracting request parameters are unmodified and non-integrated with external systems [5, 6].

There are frameworks providing combined solutions for the development and training of chatbots, for example, the framework developed by the iPavlov.ai laboratory. It combines components of rules-based approach and machine learning elements [7].

3. The main goals and purposes of chatbot
Reduce time and energy costs, processes optimization and elimination of human element’s errors are the primary aim of chatbot assistant. The data collected by the chatbot is automatically classified and systematized, it simplifies record-keeping and analytical data collection.

Chatbots are used for conducting a dialog with users and besides, for entertainment and information purposes. For example, chatbot can tell you about forecast or currency rate, make an appointment with a doctor and so on [8].

Nowadays, chatbot is able to give an answer to any question, if you load all the necessary information into it. Most importantly, a bot works quickly and unmistakably that’s because it has the ability to replace men [9, 10].

Currently, chatbots are applicable in most spheres of human life and chatbot’s goals and purposes depend on them. One of these spheres is time management. Bot assistant is able to scheduling for user: it analyzes the date and offers several variants for allocating time. That bot purpose is working as an secretary. So, we can a conclusion that the mission of any chatbot is facilitating human labor in a chosen field of activity [11].

Any field of human activity requires the results of metrological research. However, no chatbot has been created to optimize measurement performance. It could reduce the time spent on performing and processing metrological measurements. For such a chatbot, the purpose would be to reduce the time of accessing the metrological system, as well as to eliminate the appearance of human factor’s errors in mandatory measurement actions.

4. Overview of analogs
In October 2017, Yandex launched the voice assistant – Alice. The dialog system can solve any tasks, search for information in the dialog format by text or by voice input, Alice is able to answer the question imitating human speech [12].

Siri is an AI voice assistant, Siri’s development was started in 2007 by Siri Inc. It does similar tasks to Alice and some special tasks, for example, set an alarm, make a call to contacts and so on [13, 14].

As we said before, no dialog system (chatbot) has been created to help you perform the metrological measurements, so there are no any software analogs.
5. Scientific innovation. Natural language understanding voice assistant. Application of the proposed solution for metrological measurements

Metrology plays a leading role in the progress of technology. It must develop faster than other fields of science and technology because each of them requires accurate measurements for continuous improvement. Subject of metrology is extraction of quantitative information about investigated object’s features with the required degree of accuracy and reliability.

The developed dialog system is highly mobile and compact, and does not require powerful and expensive hardware. Existence of a chat and voice assistant significantly reduces the time spent on accessing the system, and also reduces the time required for execution of works. Each metrological measurement contains a number of required steps, voice chatbot assistant performs the role of an assistant for recording the completed measurements. It will be able to remind you of the next necessary action and the minimum number of measurements. All data entered manually or using voice commands will be recorded in the memory of the assistant bot and can be downloaded upon completion of all required tests.

6. Offers for the application’s architecture and functionality

It’s necessary to store certain data to develop an API (application programming interface) that processes user requests. The date a set of dictionaries a set of dictionaries that describe the user’s terms used for interaction with the application.

Application generated responses are based on existing application resource – JSON-scenario (JavaScript Object Notation). Response scenario is divided to next fragments:

- Greeting;
- Construction of the model of the measurement (enumeration of all the values);
- Recording the measurement;
- Completing the measurement;
- Checking the result.

Each fragment is a block containing the following fields:

- Name. The fragment name;
- Text. Text describing the fragment’s content;
- Condition. Condition using a fragment to generate a response.
- Switch. The switch statement can have a number of possible execution dialogue paths.
- Parameters. Fragment parameters that serve to increase the accuracy and detail of the response.

The application functions using the following algorithm:

First of all, you need to select meaningful terms from the user’s message and determine their semantic meaning. All significant semantic values are pre-encoded into information slots in the key format (semantic context), it means possible values of the semantic context. For example, metrological terminology semantic context contains terms like measurement units, spatial orientation, space coordinates, architectural and math terms, user actions.

Application recognizes the user's intent, when message information slots containing some context are defined. The dialog script is loaded according to the intent. In the scenario, possible execution dialogue paths and chatbot action depend on information slots context. «Condition» block analyses existence of information slots and their values and instructs the dialogue path.

Example, the condition below checks for information slot «Greeting», if condition is true, then dialogue «Welcome» will load, otherwise, dialogue «Measurement» will load.

```
"condition": "[greeting]",
"switch": {

"dialogId":"Dialog.Greeting"
```
Possible chatbot actions are: text message output, new dialog download, accessing to the API to generating a request.

Figure 1 shows a graphical diagram of the scenario.

![Graphical diagram of the scenario](image)

**Figure 1.** Graphical diagram of the scenario.

Scenario example:

```json
{
   "Name": "Dialog.Greeting",
   "text": "[greeting]",
   "condition": "[greeting]",
   "switch": {
      "condition": "[Measure.Begin]",
      "dialogId": "Dialog.Measure begin"
   }
},
{
   "Name": "Dialog.Measure begin",
   "text": "Enumerate model parameters",
   "condition": "[Measure.Begin]",
   "switch": {
      "condition": "{Term. Agreement}"
   }
},
{
   "Name": "Dialog.List parameters",
   "text": "Building model",
   "condition": "[Measure.Builted]",
   "switch": {
      "condition": "{Term. Agreement}"
   }
},
"parameters": {
   "name": "Model.Measure",
   "text": "What will we measure?"
},
{"name": "Measure.Value",
"text": "What size is it[measure]?

}```
For optimization of the algorithm and elimination of deadlocks in behavior of the program, you should to use the theory of automata [15]. Based on the above scenario, you can build an automaton of the Moore that describes the cycle of user interaction with the voice assistant [16]. Thanks to the automata, you can describe and simulate the program's operation according to the given scenario, and to optimize an algorithm for processing user requests. The state machine is shown on the figure 2.

![State Machine Diagram](image)

**Figure 2.** State machine.

S₀ is initial state, S₁ is waiting state for input of the next model parameter, S₂ is input parameter’s initialization state, S₃ is input parameter’s initialization error condition state, S₄ is waiting state of input parameters values, S₅ is waiting state of saving values, S₆ is error state of saving values, S₇ is status of
checking the input values, \( S_k \) is final state, \( x_0 \) is a greeting signal, \( a_0 \) is input model parameter signal, \( a_1 \) is successful initialization signal, \( x_1 \) is end signal of the model initialization, \( b_0 \) is signal of input parameter’s value, \( b_1 \) is signal of successful saving input values, \( x_2 \) is a termination signal of measurements, \( x_3 \) is signal of accuracy measurements.

Also, the scenario can be represented in the form of a canonical system of equations, presented on the formula 1.

\[
\begin{align*}
    s_1 &= s_0 \& x_0 \mid s_0 \& x_1 \mid s_2 \& a_1 \mid s_3 \mid s_7 \! \& x_3 \\
    s_2 &= s_1 \& a_0 \\
    s_3 &= s_2 \& ! a_1 \\
    s_4 &= s_1 \& x_1 \mid s_5 \& b_1 \mid s_6 \mid s_4 \& ! x_2 \\
    s_5 &= s_4 b_0 \\
    s_6 &= s_5 \& ! b_1 \\
    s_7 &= s_4 \& x_2 \\
    s_k &= s_7 \& x_3
\end{align*}
\] (1)

A mathematical simulation of the scenario was carried out by using the event-based nondeterministic finite-state machine to build an automaton model.

One of the important applications’s features is the ability to evaluate the popularity of terms used by the user. This value is called TF-IDF (Term Frequency-Inverse Document Frequency).

TF-IDF is a statistical measurement, it is used to evaluate the importance of a word in the document context, that included in the collection. The weight of a certain word is equal to the frequency using this word in the document under study and is inversely proportional to the frequency using of the word in the documents included in the collection [17].

Term Frequency-Inverse Document Frequency is using in tasks related to text analysis and information search, for example, it can be one of the criteria for the relevance of a document to a search query to calculating the measure of proximity of documents for clustering. The value is calculated by using two values:

Term frequency (tf) is the ratio of the number of occurrences of a certain word to the total number of words in the document. Thus, it is possible to estimate the word importance within the researching document. TF is calculated using the next formula 2.

\[
    tf = \frac{n}{m}
\] (2)

\( n \) is the number of occurrences of the evaluated term in the document, \( m \) is the total words count in the document.

The inverse document frequency (idf) is the inversion of the certain word frequency which occurs in documents in the collection. IDF helps to less the weight of commonly used words decreases. There is a single IDF value for all unique words in a certain document collection. This value is calculated using the next formula 3.

\[
    idf = \log\frac{m}{n}
\] (3)

\( m \) is the total documents count in the collection, \( n \) is the total documents containing this term.

The choice of the base of the logarithm in the formula does not matter, as changing the base causes the weight of each word to change by a constant, it does not affect the ratio of weights [18].

Therefore, the TF-IDF is the result of two factors: TF and IDF. If a word has a high frequency of use in the current document and a low frequency of use in the collection documents, then the word has the highest weight in TF-IDF [19, 20].

7. Describing application work

Interaction with the dialog service occurs in the following sequence:
- The user starts the application.
- The system goes to the initial state and analyzes incoming requests.
- The entered message is processed and analyzed using the features described above. The entering message is parsed into separate sentences. The words in each sentence analyzes morphologically. Also, each word evaluates according to the TF-IDF metric. The words with the highest rating marks as key words. Response on user’s request is based on this calculating.
- Keywords relates to different types of activities. Matching keywords to activity types are stored on the server in JSON-format, and represent a key: value pair.

UML diagram of performing user’s request is represented on figure 3.

![UML-diagram of performing user’s request](image)

**Figure 3.** UML-diagram of performing user’s request.

For the correct work of the application, it is suggested to use the next services, which are combined in a chain. Each service is processing the request sent by the previous service. Also, each service may return error status as a response to the service that delegated.

- Dialog service. The task of service is an exchanging requests and response with the user.
- Morphological analysis service. The main task of the service is a convert words into their initial form.
- Service of semantic analysis of sentences. The task of this service is a split the sentence by words and determine the meaning of each word in the context of the sentence.
- Importance evaluation service. The task of service is a select the most important words from the user’s request for building the most accuracy response.
- Evaluating tonality service. The main task of this service is a select negative and obscene words from the user's message and a process them properly. Figure 4 shows the scenario diagram.

![Scenario diagram](image)

**Figure 4.** Scheme for dialog of separate scenario.
An example of such a scenario is given below.

```json
{
  "Name": "swearing",
  "Category": "Profanity",
  "Description": "Profanity",
  "Questions": [
    {
      "text": 
      "\{term.human\}\{words.swearing\}"
    },
    "Answers": [
      {
        "Text": "Please, don’t use words such as, '{words.swearing}’,",
        "Probability": 100
      }
    ]
  ]
}
```

• Response generation service. The main task of this service is generating a response to the user based on the result of previous services. The mechanism works in the following way. After normalization the incoming message and highlighting the keywords, the application analyzes all the scenarios of the dialog development. When most corresponding scenario is founded, the service builds response based on this scenario. The scheme of the scenarios is described on the figures 5 and 6.

![Figure 5. Schema of scenarios.](image)

![Figure 6. Structure of the Question block.](image)

An example of scenario of greeting is given below.

```json
{
  "Name": "greeting",
  "Category": "greeting",
  "Description": "Greeting - a action, word, phrase, written message (and their combination), or other ritual for bringing a person (group of people) into contact with another person (group of people).",
  "Questions": [
    {
      "text": 
      "Hello",
      "Hello {self}",
```
8. Conclusion
Rule-based approach is one of the most popular modern approaches to developing a chatbot was investigated. Was described the features of rule-based mechanism and reviewed similar assistant bots.

As a research result, we described variants for developing a chatbot that allows you to perform and record metrological tests based on data entered in a chat or using a voice assistant.

Acknowledgments
The work was supported by the RFBR grant "Competition for the best projects of basic scientific research", grant No. 19-07-00516 A.
References

[1] Jurafsky D and Martin J H 2008 Speech and Language Processing (USA: Prentice Hall)
[2] Dzhanarsanam S 2019 Hands-On Chatbots and Conversational UI Development ed D A Movchan (Moscow: DMK-Press)
[3] Smyslova L V 2018 Chatbot as a modern means of internet communication Young Scientist 9(195) 36-9
[4] Uraev D A 2019 Classification and methods of creating chatbot applications International scientific review
[5] Baryshnikova N Y 2018 Organization of interactive information systems New information technologies in science (Ufa: OMEGA SCIENCES Limited Liability Company) p 6-8
[6] Matveeva N Y and Zolotaryuk A V 2018 Technologies for creating and using chatbots Scientific notes of young researchers
[7] Menyashev R E, Smirnova E V and Susev V V 2019 The possibilities of using chatbots in Internet of Things technologies Problems of modern science and education
[8] Tugusheva N A 2017 Using chatbots in various areas of everyday life Young Scientist 21(155) 36-3
[9] Provocar A I and Klochkho K A 2013 Features and problems of virtual communication using chatbots Scientific works of the Vinnytsia national technical university 22
[10] Nikitin M N 2016 Prospects for the use of chatbots in logistics Materials and Methods of Innovative Research and Development (Ufa: OMEGA SCIENCES Limited Liability Company) p 84-6
[11] Financial Telegram Bot in 30 Minutes with Market Data Retrieved from: https://habr.com/ru/company/exante/blog/318272.
[12] Alice How Yandex Teaches Artificial Intelligence to Talk to People Retrieved from: https://habr.com/ru/company/yandex/blog/339638/
[13] What is Artificial Intelligence and how is it Trained to Perform Tasks that People are Used to Coping with? Retrieved from: https://tass.ru/ekonomika/7691775
[14] Basalaeva M A 2020 Analysis of modern approaches to the use of digital assistants Mathematics and Computer Science in Education and Business (Moscow: Aegitas) p 51-5
[15] Pashchenko D V, Martyshkin A I and Trokoz D A 2020 Decomposition of Process Control Algorithms for Parallel Computing Systems Using Automata Models International Russian Automation Conference 839-45
[16] Martyshkin A I, Salnikov I I, Pashchenko D V and Trokoz D A 2018 Associative Co-processor on the Basis of Programmable Logical Integrated Circuits for Special Purpose Computer Systems Global Smart Industry Conference (GloSIC 2018) (Chelyabinsk) 1-5
[17] Mihajlov D V, Kozlov A P and Emefyanov G M 2017 TF-IDF Measure, Word Connection Strength, and Formation of Knowledge Representation Units in Open Tests Mathematical Methods of Pattern Recognition 18(1) 136-7
[18] TF-IDF Retrieved from: https://ru.wikipedia.org/wiki/TF-IDF.
[19] Hiemstra Djord 2000 A probabilistic justification for using TF×IDF term weighting in information retrieval Heidelberg International Journal on Digital Libraries 3(2) 131-9
[20] Yashina A G and Prozorov D E 2014 Model of information search of speech documents by text query based on phonemic transcription and TF-IDF measures Algorithms Methods and Data Processing Systems 1(26) 69-78