Development of an assistant chat-bot with voice control elements for teaching how to work with the CNC system

Ramil Nezhmetdinov1*, David Malykh1, Ramilya Nezhmetdinova1, and Ilya Kovalev1

1MSUT "STANKIN", 127055, Moscow, Russia, Vadkovskiy per., 1

Abstract. The article discusses the principle of building a voice assistant for learning how to work with the "AxiOMA Control" CNC system. Methods for solving the problem of designing voice assistants are compared, including: speech recognition, dialogue mode and speech synthesis mode. Developed own methodology applicable to the learning process. On the basis of the methodology, a block diagram and algorithms for the operation of an intelligent assistant have been developed. A program is proposed for generating images with hints, as well as a set of files for generating sound files describing the operation of the CNC system. Testing was carried out and program changes were made to the project.

1 Introduction

The development of information technology leads to the automation of many industries - partially or completely replacing a person in routine tasks. Thus, this development has also affected the issues of education: in the era of the Internet, each teacher can once record a course of his lectures and put them in free access - this allows you to approach the issue of education more flexibly, without tying it to a specific date and time [1, 2].

But not every student is able to independently understand the existing topic, which leads to the emergence of various kinds of questions when mastering the discipline. These questions are very often of the same type, that is, routine for the teacher, and some of them can be described in advance to save time. Modern technologies allow you to create intelligent dialogue systems, in other words, chat bots that allow you to build a dialogue in a limited number of questions.

A chat bot is a kind of autoresponder with predetermined topics of conversation and the structure of the dialogue. It allows you to automate the search for the necessary information by a person.

This article describes the approach that is necessary to solve the problems of teaching students at the initial level to understand the "AxiOMA Control" CNC system, allowing them to conduct a dialogue with the program in the natural language of symbols or audio signals, providing text, photo and audio signal (speech) as a response [3].

* Corresponding author: r.nezhmetdinov@stankin.ru

© The Authors, published by EDP Sciences. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).
During development, an analysis of various models of speech recognition was carried out. A Hidden Markov Model (HMM) is a stochastic model that is used to describe a sequence of events or data that is in a temporal sequence [4]. The use of HMM in human speech recognition is based on the statistical analysis of acoustic signals that are generated during the pronunciation of words. HMMs are used to model the sequence of phonemes (sound units of a language) that are commonly used to distinguish word meanings. The advantages of HMMs in speech recognition are their ability to effectively model the temporal and spectral variations of the speech signal, powerful mathematical tools and flexible topology.

For acoustic-phonetic modeling of a speech signal, various models are used, including models of artificial neural networks (ANNs), which are based on biological models of the nervous system. To date, there are a fairly large number of neural network models, but there are not so many practically implemented ones - they all consist of layers. The most popular learning procedure is the backpropagation algorithm, described by Frank Rosenblatt in 1959. It is based on gradient descent, which allows you to minimize the difference between the received vector and the desired one. The use of various multilayer perceptron learning algorithms improved the recognition of short-term acoustic-phonetic units, including phonemes, but did not lead to an equally significant improvement in the recognition of long acoustic data sequences that are used to represent words [5].

In 1986, American scientists from Carnegie Mellon University, James Beck and Michael York, proposed using a combination of HMM and NN to improve the accuracy of speech recognition. They created a hybrid speech recognition model that combined HMM to model the time dependence of speech sounds and NN to highlight phoneme features. Hybrid Speech Recognition uses a neural network to predict the probability distribution that a particular sound (phoneme) will follow another sound, and then HMM is used to predict the most likely word or phrase based on that distribution [6]. One of the main advantages of the hybrid speech recognition method is the high accuracy achieved by combining two different methods [7]. Also, the hybrid approach allows you to cope with different accents, intonations and pronunciations of words, which makes it one of the most effective speech recognition methods at the moment.

Table 1 provides a brief description of the methods that solve the problem of modeling intelligent voice assistants. Depending on the operating conditions, the developer can choose a suitable set of methods.

**Table 1.** A brief description of the methods that solve the problem of modeling voice assistants

<table>
<thead>
<tr>
<th>Task</th>
<th>Method</th>
<th>Short description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recognition speeches</td>
<td>Hidden Markov Models</td>
<td>Powerful mathematical apparatus; effective modeling of a speech signal (temporal and spectral); flexible topology; deep practical study</td>
</tr>
<tr>
<td></td>
<td>Neural networks</td>
<td>Carries out discriminant learning between speech units; can find the optimal combination of constraints for classification; uses parallel learning</td>
</tr>
<tr>
<td></td>
<td>Hybrid models: ANN and HMM</td>
<td>Improved model accuracy, discriminant power, system dimensionality reduction, flexibility, complementarity</td>
</tr>
<tr>
<td></td>
<td>Ranking Models</td>
<td>Working on previously known commands based on a limited list of keywords. They are much easier to create,</td>
</tr>
<tr>
<td>Conducting a dialogue</td>
<td>but the practical use of such chatbots is limited to keywords.</td>
<td></td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Generative Models</td>
<td>Self-learning models based on Machine Learning algorithms and natural language understanding methods. Flexible in communication, but difficult to track errors.</td>
<td></td>
</tr>
<tr>
<td>Synthesis speeches</td>
<td>Selective speech synthesis</td>
<td></td>
</tr>
<tr>
<td></td>
<td>It is a kind of concatenative speech synthesis (in the process of speech signal synthesis, pre-made sound recordings of natural speech are used). It is currently the main technology for automatic speech synthesis.</td>
<td></td>
</tr>
<tr>
<td>Statistical parametric synthesis</td>
<td>With this approach, an attempt is made to machine-learn the system on the available speech data in order to obtain a model for matching speech characteristics. Works well with common words, but poorly with narrow-profile ones.</td>
<td></td>
</tr>
</tbody>
</table>

### 2 Methodology for analyzing user requests using intelligent systems

A method for creating voice assistants was developed to solve the problem. Which contains 6 steps, each of which gives a certain result (Fig. 1).

![Diagram](image)

**Fig. 1.** Method of creating voice assistants for learning how to work with CNC.

Step 1 - Analysis of user interfaces. This step considers the conditions of the application of the system, the necessary corresponding user interfaces. So, for example, sound generation as a means of interaction with the end user can be hampered by various kinds of interference (for example, noise in the shop). In such conditions, the usual way of input - output of information - through the screen and keyboard, is more suitable.

Step 2 - Analysis of input, output information. By analogy with the conclusion, not all data can be acoustically described: either due to the complexity of the data itself, or they are too large, the rules of acceptable words - descriptions, and the user's vocabulary are also imposed. Therefore, it is worth considering in advance both the complexity of the described object and the intuitive clarity of what can be described and how it can be done.
Step 3 - Selecting an algorithm for issuing a response. There are two approaches: generative and ranking. The ranking method has a predetermined set of input commands and output - responses to them. The generative method is based on neural networks and does not have a strictly regulated set of input and output commands. Therefore, they are used in a hybrid form, as a question classifier (from predefined ones), which gives flexibility to the input data, but the response to the request is always regulated, which removes the problem of issuing incorrect information. It is also worth considering the amount of data on the lines of dialogue. If the options for the answers given by the system can still be regulated and accurately described, then the options for the questions asked by the user can be very wide. And here there can be two options: the use of a rigid framework for conducting communication - the system itself offers options for the user's request and gives ready-made answers, or the use of a "semi-rigid" framework - the user himself can issue a request in a free speech form, and the system must attribute it to one of the already ready-made options. The problem with this option is the amount of data available for modeling communication, machine learning methods are used here, and they need a lot of data, otherwise the model will be incapacitated.

Step 4 - Choice of frameworks and libraries. This step defines the architecture of the system, the platform it will run on, and the security issues of the system. Ready-made solutions can be either entire software modules or libraries used to design these modules.

Step 5 - Module testing and analysis of the results. At this stage, the correctness of issuing an answer is checked, which is based on the completeness of the description of the scenario recognition algorithm: either by keywords, if there is not a large amount of dialogue conducting data, or by the degree of training of the neural network that identifies the dialogue line.

Step 6 - Making adjustments. If necessary, adjustments are made to the algorithms that implement various models, as well as to data that could be incorrectly labeled or incompletely filled.

On the basis of the developed methodology, in this work, a voice assistant was implemented for teaching students the "AxiOMA Control" CNC system [8-10]. The assistant will have two ways of input - output of information: voice, and standard (monitor, keyboard). Input and output information can be presented both in the form of a photo (screenshot of the program) and in the form of a voice description (buttons that the user sees). The model for issuing an answer in learning tasks can be either ranking or hybrid.

3 Preparation for development

The project of an intelligent voice assistant in this work is aimed at solving problems of learning optimization. His tasks will include solving the most common problems for students that arise during the study of the "AxiOMA Control" CNC system.

"AxiOMA Control" is a universal numerical control (CNC) system used to solve technological problems on various types of machine tools, including new models of technological equipment. In addition, the CNC system is equipped with the means to connect to modern workshop networks [11, 12].

The main problems in learning this CNC system include: orientation inside the CNC (file operations, setting, creating a tool, enabling 3D mode, selecting a channel, etc.), working with system errors (3D mode does not work, tool not found, coordinates exceeding limits, etc.), working with errors in G code - (no security string, incorrect coordinate system, syntax error, etc.) [13-15].

These are the main points that are emphasized when creating a voice assistant.

Python was chosen as the development language, Python Imaging Library (PIL) is a library for working with images in the Python programming language. With it, you can load
and save images in various formats, resize them, apply filters and effects, and create and edit graphic elements.

Tkinter is chosen for the user interface library. Libraries were also used: "speech_recognition" is a library that solves the problem of recognizing natural human speech, "wave" is a library that provides a convenient interface for writing and reading WAV audio format and "playsound" is a cross-platform module that can play such audio files like WAV, AU, AIFF, MP3, CSL, SD, SMP and NIST/Sphere.

The whole intelligent assistant system consists of 4 modules: text input/output system, sound input/output system, dialog system, handler/database. The sequence diagram in the figure best reflects the process of interaction between the components of the system.

A structural diagram of the interconnection of components was developed (Fig. 2), which is divided into an input system, a dialog system and an output system. The input system includes input means such as a keyboard and a microphone. If the user uses a microphone, then the audio stream must first be converted to text. In this case, it happens on a remote server. This is followed by a dialog system, which actually consists of the main script of the program, which includes three text files: name, hello, keyWord. The last module is called "Inference System". It is responsible for displaying information stored in scripts.

![Structural diagram of the interaction of system components.](image)

The Structural diagram shows four scenarios, but their number is not limited. The main problem is in their differentiation, which comes down to the selection of non-overlapping keywords. On the one hand, this is a disadvantage, on the other hand, it is an advantage, which consists in the ease of setting up the entire system for teachers who are absolutely not related to programming. Also on the diagram are audio files. They are not mandatory, and can be applied flexibly depending on the scenario.

### 4 Development of a voice assistant for learning how to work with CNC

After starting the program (Fig.3), the user immediately sees two main areas of the window that opens: text and window interface. The text interface displays a greeting from the chatbot, an example of the user addressing it, as well as script options that the user can choose himself. The window interface displays photos with arrows indicating the buttons to be pressed, as
well as additional information for interacting with the "AxiOMA Control" CNC system [16-18]. To switch photos, there are three buttons at the bottom: forward, back and go to the beginning. So the user will be able to consistently get acquainted with all the stages of work in the program, and dwell in more detail on what interests him more.

![Start system window with main menu of bot.](image3)

Fig. 3. Start system window with main menu of bot.

To interact with the bot (display the desired script), there are several ways to enter information: from the keyboard or using the user’s voice. The audio stream is processed by a hybrid model of Markov chains and neural networks and converted into plain text information. Data entered from the keyboard is immediately textual information. On the main screen, at the bottom left, there are additionally two buttons: for voice input and voice output of information.

Each scenario has a specific set of pictures and sound files to help you learn (Fig.4).

![Folder with data for chat bot voice assist.](image4)

Fig. 4. Folder with data for chat bot voice assist.
There is a native chatbot file system as well as methods for adding new scenarios. Each scenario is a separate folder, there is always a main script called "help", and the teacher can add additional scripts to suit his needs. The "help" script contains several photos and text files: name, "hello", "KeyWord", "scenario". The file name contains the name of the chatbot, it can be arbitrary (Fig 5).

![Fig. 5. Main scenario files.](image)

The "hello" file contains a greeting that is displayed once when the bot is turned on. The "KeyWord" file contains the keywords and script names for them. The scenario file contains a script that will be displayed in the text field of the interface each time it is called, it should contain information - a hint for interacting with the bot. You need to add photos to this folder, each of which should be numbered from one to the last photo in order, without skipping numbers. You also need to add a text file called scenario, and write in it the text corresponding to the set of photos. In addition, you can add audio files, also numbering them one by one like the photo, but some of them can be skipped, the main thing is that the audio recording number matches one of the photo numbers.

Scenario development is the main part of the preliminary preparation of the voice assistant. Depending on the discipline being studied, the teacher should choose the main lines of dialogue regarding the most frequently asked questions. There are currently 11 scenarios available.

For example, "Take a quick look at "AxiOMA Control" [19]. In this scenario, the student is shown the main elements of the interface with which he must work (Fig.6).

![Fig. 6. First Scenario for teaching to work with CNC.](image)
scenario.txt:
You are presented with three windows, in two of them you will work:
1. The main window - in it you will spend 99% of the time
2. Virtual machine panel - from this window you need only three buttons: Reset, Start, Stop
3. Kernel operation window (WinNc) - you don't need it at all, but you can't close it, otherwise the program will close.

Another example. The user activates the voice input mode and asks the question: "I want to select a cutting tool" [20]. Using voice recognition and keywords, the system determines the number of the required script and opens the "Tool Management" screen and explains how to select a tool and transfer it to the right tool magazine (Fig. 7).

![Fig. 7. Scenario for teaching to work with Tool Management.](image)

5 Conclusion
As a result of consideration of this topic, we can say that an analysis of existing approaches to solving the problem of designing intelligent dialogue systems was made. Also, the following were developed: a methodology for designing intelligent dialogue systems, algorithms for the operation of the main components of the system, and their software implementation. Further, an automated system for conducting a dialogue at the level of the user interface and program modules was designed. The system has been tested for various errors and exceptions. The process of working with this software module was also described in detail.

At the moment, the system continues to develop. It is expected that more advanced natural language processing algorithms will be added to improve the learning ability of the chat bot.

Acknowledgment
This work was supported by the Ministry of Science and Higher Education of the Russian Federation under project FSFS-2023-0004. The study was carried out on the equipment of

References


