Investigation and Development of the Intelligent Voice Assistant for the Internet of Things Using Machine Learning

Polyakov E.V., Mazhanov M.S., Rolich A.Y., Voskov L.S., Kachalova M.V., Polyakov S.V.

Abstract— **Artificial intelligence technologies are beginning to be actively used in human life, this is facilitated by the appearance and wide dissemination of the Internet of Things (IoT). Autonomous devices are becoming smarter in their way to interact with both a human and themselves. New capacities lead to creation of various systems for integration of smart things into Social Networks of the Internet of Things. One of the relevant trends in artificial intelligence is the technology of recognizing the natural language of a human. New insights in this topic can lead to new means of natural human-machine interaction, in which the machine would learn how to understand human's language, adjusting and interacting in it. One of such tools is voice assistant, which can be integrated into many other intelligent systems.**

In this paper, the principles of the functioning of voice assistants are described, its main shortcomings and limitations are given. The method of creating a local voice assistant without using cloud services is described, which allows to significantly expand the applicability of such devices in the future.

Index Terms— **NLU, TTS, NER, ASR, Internet of Things, IoT, Voice Assistant, smart things, machine learning,**

I. Introduction

TODAY the development of artificial intelligence (AI) systems that are able to organize a natural human-machine interaction (through voice, communication, gestures, facial expressions, etc.) are gaining in popularity. One of the most studied and popular was the direction of interaction, based on the understanding of the machine by the machine of the natural human language. It is no longer a human learns to communicate with a machine, but a machine learns to communicate with a human, exploring his actions, habits, behavior and trying to become his personalized assistant.

The work on creating and improving such personalized assistants has been going on for a long time. These systems are constantly improving and improving, go beyond personal computers and have already firmly established themselves in various mobile devices and gadgets. One of the most popular voice assistants are Siri, from Apple, Amazon Echo, which responds to the name of Alex from Amazon, Cortana from Microsoft, Google Assistant from Google, and the recently appeared intelligent assistant from Yandex, under the name "Alice" [1-3].

Section I, II presents a brief introduction to the architecture and construction of voice assistants. Section III provides an example of the work of a voice assistant based on Alice, a product from Yandex. Section IV describes the shortcomings of existing voice assistants and how to solve them. tools necessary development of such systems. Section V describes the methods for developing and training an assistant using various machine learning algorithms, and gives a comparative evaluation of the learning ability of algorithms. The main goal of this work is to build a local voice assistant that does not depend on various cloud technologies and services, which would allow using it to solve various specific problems.

II. Intelligence Voice Assistant Technologies

Each company-developer of the intelligent assistant applies his own specific methods and approaches for development, which in turn affects the final product. One assistant can synthesize speech more qualitatively, another can more accurately and without additional explanations and corrections perform tasks, others are able to perform a narrower range of tasks, but most accurately and as the user wants. Obviously, there is no universal assistant who would perform all tasks equally well. The set of characteristics that an assistant has depends entirely on which area the developer has paid more attention. Since all systems are based on machine learning methods and use for their creation huge amounts of data collected from various sources and then trained on them, an important role is played by the source of this data, be it search systems, various information sources or social networks. The amount of information from different sources determines the nature of the assistant, which can result as a result. Despite the different approaches to learning, different algorithms and techniques, the principle of building such systems remains

The article was prepared within the framework of the Academic Fund Program at the National Research University Higher School of Economics (HSE) in 2017 — 2018 (grant № 17-05-0017) and by the Russian Academic Excellence Project «5-100».

Polyakov E.V. (e-mail: epolyakov@hse.ru), Mazhanov M.S. (e-mail: msmazhanov@edu.hse.ru), Rolich A.Y. (e-mail: arolich@hse.ru), Voskov L.S. (e-mail: lvoskov@hse.ru) are affiliated with the National Research University Higher School of Economics, Russian Federation.

Kachalova M.V. is affiliated with University of Mannheim, Germany (e-mail: mkachalo@mail.uni-mannheim.de).

Polyakov S.V. is affiliated with the Moscow Aviation Institute, Russian Federation (e-mail: s.polyakov@mai.ru).

978-1-5386-3498-1/18/$31.00 © 2018 IEEE

approximately the same. Figure 1 shows the technologies that are used to create intelligent systems of interaction with a human by his natural language.

The main technologies are voice activation, automatic speech recognition, Teach-To-Speech, voice biometrics, dialog manager, natural language understanding and named entity recognition [3-7].

VOICE TECHNOLOGY	BRAIN TECHNOLOGY
Voice Activation	Voice Biometrics
Automatic Speech Recognition (ASR)	Dialog Management
Teach-To-Speech (TTS)	Natural Language Understanding (NLU)
	Named Entity Recognition (NER)

Fig.1. Technologies for constructing intelligent systems of interaction with a human by natural language.

III. Features of Yandex Voice Assistant "Alice"

Let's consider how voice assistants are basically arranged on an example of "Alice" from the company Yandex.

"Alice" is an intelligent assistant for smartphones and personal computers, which allows to solve common tasks of users, such as searching information on the Internet, finding places on the map, routing routes, reporting weather forecast. In this case, "Alice" can support the conversation, entertain the user, etc. To do this, "Alice" uses the cloud funds of the company Yandex, to which it refers via the API through the Internet. The scheme of work is presented in Figure 2.

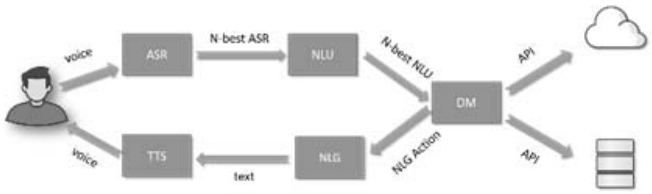

Fig.2. The scheme of work of the voice assistant "Alice"

At the first stage, activation occurs, for example, by pronouncing a key phrase. The assistant constantly listens around the surrounding sounds, analyzes the presence of the key phrase and, if it is recognized, goes into the active mode. Next, the user says the text, which can explain to the assistant what the user wants to do. The automatic speech recognition system turns the text into N-best hypotheses of what the user said. Then the natural language understanding system turns the text into N-best options for understanding the user's phrase, then the dialogue engine interprets and classifies these phrases and determines what needs to be done based on the information received. For example, contact various services for information. After obtaining the necessary data, the system performs a process of returning information to the user, i.e. The natural language generation system generates text for the user's response, then the Voice-to-Speech (Text-To-Speech) system generates sound information based on the trained models, which is announced to the user as a response. In addition to a response, any action can also take place on a mobile phone or computer, eg. Start the application or search for information in the search engine.

One of the important parts of the voice assistant from the point of view of the functional is the dialogue manager. There are simple scripts that can be immediately extracted from the NLU model and reproduced via NLG. And more complex scenarios, which are based on the concept of form. In principle, the form repeats the form of a conventional user interface (UI), where there are mandatory fields for filling and optional (Figure 3 a). In such scenarios, the form filling approach is used, i.e. in the context of the dialog, the form is filled with the necessary answers, and these answers can be filled both by the user and the system itself based on information that it can get from the user. (Fig. 3b). The filling process is also intelligent and the system itself can fill part of the field itself. After filling out the form is sent for processing, where a decision can be made about an exact answer to a user's request or switching to a neural dialog.

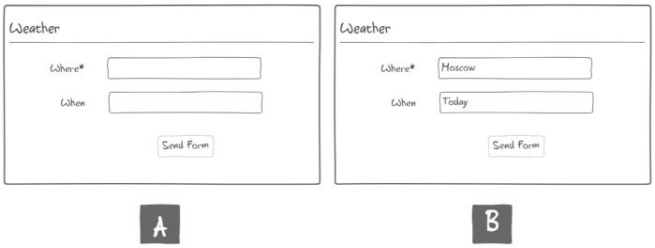

Fig.3. Form filling with the dialog manager

The main problems encountered by developers of such systems are problems of classification of scenarios, isolation of semantic objects (the need to communicate with geodatabases of data and refine the names of objects, access to other APIs), maintenance of the context, problems of ellipses and coreference.

IV. Disadvantages Of Existing Voice Assistants

Even though such assistants have existed for quite a long time, they have not been widely disseminated, due to the existence of several restrictions and areas where they cannot be applied. The main shortcomings include: focus on solutions to common problems, dependence on the Internet and cloud services, the complexity or in most cases the inability to integrate with third-party services, the insecurity of personal data [6].

Recently there has been a rapid increase in the popularity of voice assistants. They are beginning to be used in various fields, the most promising of which now is assistants for smart home systems. But their shortcomings and limitations do not allow to apply them in areas where dependence on various network infrastructure is unacceptable, for example, in medicine or in the security sphere, as well as in narrowly focused areas where application of general rules cannot solve existing problems.

It is already obvious that this area is the most promising,

especially when viewed from the point of view of the ecosystem of the future artificial intelligence, which leads to the understanding that universal systems can not cover the necessary need in different areas, and therefore the creation of more specialized personal assistants with narrower tasks and less demanding of the infrastructure, which can later be combined into a huge ecosystem is an urgent task.

Given the shortcomings of existing systems, a local voice assistant system was developed, trained to perform specific tasks, solving the problems discussed above. The following sections will describe how to create such a system.

V. Development Of Intelligent Voice Assistant For A Specific Problems Of Interaction

To work with voice recognition, it is advisable to use existing systems. For example, the PocketSphinx project. In the developed system, it was he who was chosen, as the main means of voice recognition.

PocketSphinx is a tool for automatic voice recognition [9], which works well on various low-power embedded systems, such as Raspberry Pi, and it is also cross-platform, which is the reason for choosing this framework [8].

To generate the voice, the most used Festival engine was chosen, which runs on Linux operating systems and has quite good characteristics of voice generation.

The next stage, on which the intellectual work of the voice assistant is carried out, is the creation of a system for recognizing the natural language of a person, i.e. recognition of intentions.

For the system that can understand the user, we used training intellectual algorithms based on machine learning methods. But before you teach the system to understand the user, you need to perform several stages of data preparation.

Since in our case the output characteristics can be a set of different classes (that is, n-best hypotheses of the user's intentions, the task will be a multi-class classification task where one label can contain labels from different classes. For example, as a response can return "on; light; bathroom", where each part of the answer belongs to one of several classes.

Collecting the necessary data is the most important process that will allow you to most accurately predict the results. In Fig. 4 shows the process of preparing data for training.

In our work, we created a specialized system for managing a smart home. Considering this, the given examples of data sets will correspond to the training of the model for such systems. For any other tasks, the principle will remain unchanged, only the data sets will change.

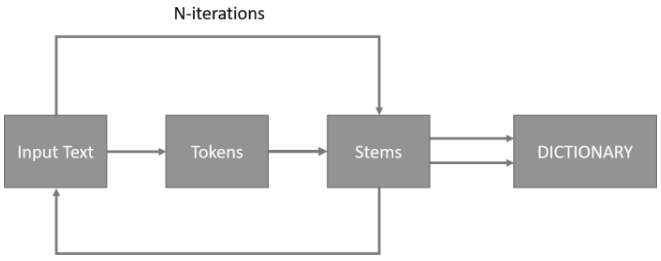

Fig.4. The process of preparing data for training

To collect data, you need to create a table of synonyms and different variants of pronunciation of keywords, according to which the system will be able to build forecasts. Words and synonyms are indicated in the format in which they are commonly used in colloquial speech. A brief example of a table of synonyms is shown in Fig. 5.

Switch on			
Turn on	Activate	Put	
Switch off			
Turn off	Cut out	Shut off	Disconnect
Light			
Lighting	Illumination	Lamp	Lantern
Split system			
Air conditioning	Cooling	Freeze	
Check			
Verification	Review	Check out	
Mail			
Mailbox	Postbox	Letter-box	

Fig.5. Table of synonyms

After the definition of synonyms, a list of answers to be predicted based on the input data is determined. The predicted answers can be in any convenient form for processing in the application. An example of the answer list is shown in Fig. 6.

On	Off	Light	Split system	Check	Mail

Fig.6. Glossary of answers

The next step is to build a vocabulary from which a training sample is created. Since most of the machine learning algorithms operate on numeric data, we must match each word in the dictionary with a unique number within the dictionary. It is also advisable to reduce the size of the dictionary with the help of stamping. On large volumes of data and different variants of pronunciation, this method allows one to reduce the number of different words by an order of magnitude. This ultimately leads to a reduction in data and will improve the quality of recognition. The list of answers should also be marked with unique numeric identifiers.

VI. Experiment and Training Sample Investigation

To determine what machine learning algorithms are appropriate to use in this task, let's look at the data distribution in the resulting training sample (Figure 7). From the resulting image, the data has a normal distribution. For the sake of simplicity, we will not analyze the data for emissions. From this we can conclude that our problem can be easily solved without using complex classification algorithms, such as: naive Bayesian classifiers, decision trees, random forest.

For training, we used the most popular library of machine

learning Scikit-Learn for the programming language Python.

Since many classification algorithms do not support multi-class classification in pure form, we will use a single classifier to train one sample and several answers. Thus, for n-binary answers we will use n-classifiers. In Scikit-Learn, this is solved using the MultiOutputClassifier object

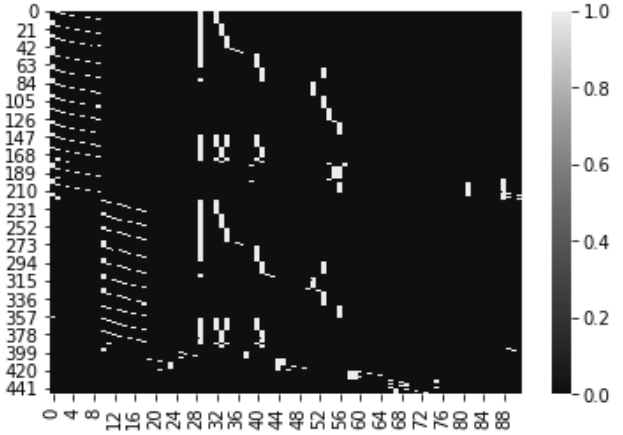

Figure 7. The data distribution matrix in the training sample

About 450 training data were used in this sample. In the case of a small set of data, it is advisable to perform a Leave-one-out cross-validation when the sample is divided into as many parts as the experiments were conducted, and the entire sample except one is used for training, and one for validation. This is quite a costly operation in terms of time and resources, but it allows the most accurate estimate of the model on small amounts of data, and especially when it is not possible to separate 30% of the test sample. The results of the experiment can be seen in Figures 8-12.

Summarizing, we can conclude that the best results of learning on a small amount of data showed the algorithm "Decision tree" with 93% of correct answers to cross-valency with the parameters max_depth = 14, max_features = 91, less able to learn the "Polynomial Bayesian Classifier" with the correctness - 81% and the lowest quality in the "k-nearest neighbors" method with a correctness of 73%.

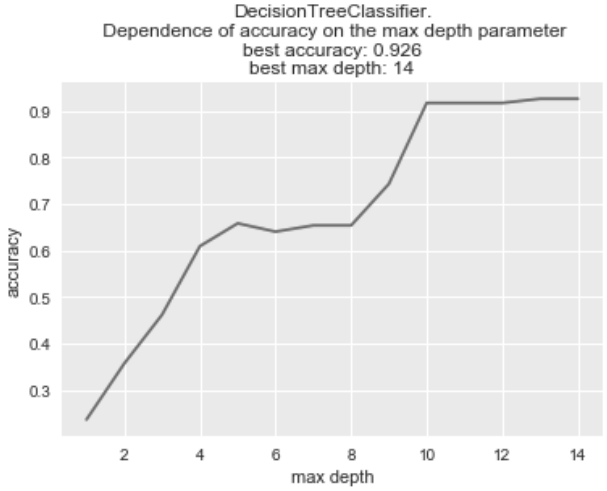

Figure 8. Algorithm "Decision tree". The investigated parameter max_depth

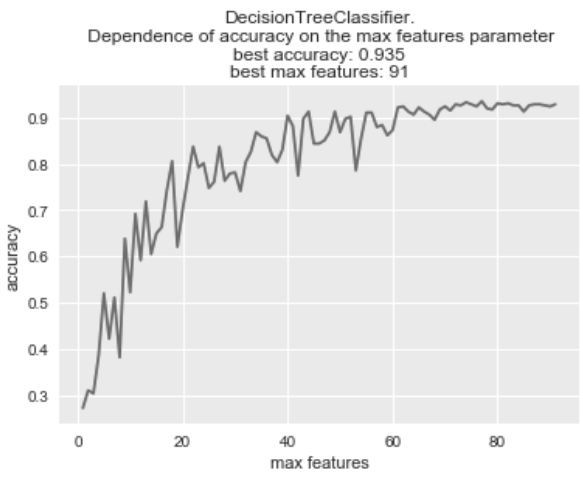

Figure 9. Algorithm "Decision Tree". The investigated parameter max_features

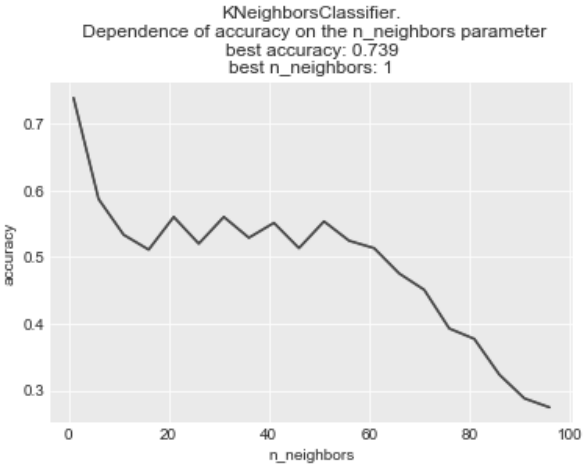

Figure 10. Algorithm of "k-nearest neighbors". Investigated parameter n_neighbors

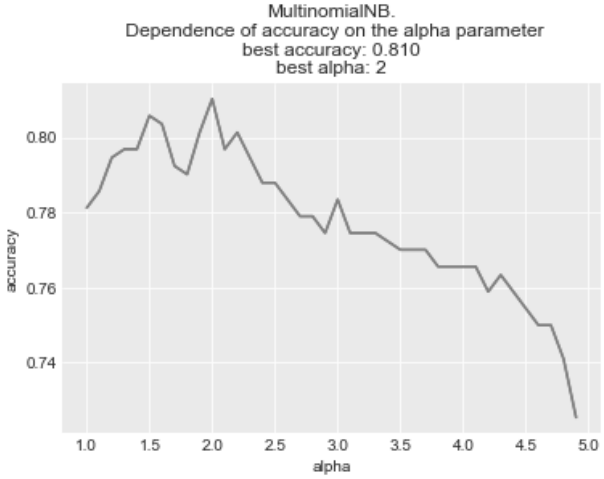

Figure 11. Algorithm "Polynomial naive Bayesian classifier". The investigated parameter “alpha”.

VII. Conclusion

The principles of voice assistants which are currently represented on the market, were conducted in the work. The main shortcomings from relevant research were identified. A method of solving these shortcomings was proposed. In the progress, a voice assistant was built and trained. Withal, evaluation of the algorithms' learning ability for recognizing intentions was performed.

As a result, the outcome is that for the presented data set (about 450 samples), the best result shows the algorithm of "Trees of Solutions" with an accuracy of 93%. Besides, the study revealed the creation and the use of voice assistants is not limited only to cloud services.

Moreover, the use of local systems allows to expand the range of tasks in which they can be applied in IoT and IIoT systems, smart home systems, healthcare, security and systems with an increased level of confidentiality, where the use of cloud technologies can be difficult.

References

[1] Dempsey P. The teardown: Google Home personal assistant //Engineering & Technology. – 2017. – Т. 12. – №. 3. – С. 80-81.

[2] Chung H. et al. Alexa, Can I Trust You? //Computer. – 2017. – Т. 50. – №. 9. – С. 100-104.

[3] López G., Quesada L., Guerrero L. A. Alexa vs. Siri vs. Cortana vs. Google Assistant: A Comparison of Speech-Based Natural User Interfaces //International Conference on Applied Human Factors and Ergonomics. – Springer, Cham, 2017. – С. 241-250.

[4] Arriany A. A., Musbah M. S. Applying voice recognition technology for smart home networks //Engineering & MIS (ICEMIS), International Conference on. – IEEE, 2016. – С. 1-6.

[5] Caranica A. et al. Speech recognition results for voice-controlled assistive applications //Speech Technology and Human-Computer Dialogue (SpeD), 2017 International Conference on. – IEEE, 2017. – С. 1-8.

[6] Assefi M. et al. An experimental evaluation of apple siri and google speech recognition //Proccedings of the 2015 ISCA SEDE. – 2015.

[7] Natural Language Understanding Lecture 10: Introduction to Unsupervised Part-of-Speech Tagging // www.inf.ed.ac.uk URL: https://www.inf.ed.ac.uk/teaching/courses/nlu/lectures/nlu_l10_unsuptag1.pdf

[8] Caranica A. et al. Speech recognition results for voice-controlled assistive applications //Speech Technology and Human-Computer Dialogue (SpeD), 2017 International Conference on. – IEEE, 2017. – С. 1-8.

[9] CMU Sphinx Toolkit: http://cmusphinx.sourceforge.net.